

**Universidade do Minho**  
Escola de Engenharia

João Miguel Timóteo Antunes

## **Recolha e Análise de Dados de Contactos Físicos e Sociais numa Rede Tolerante a Atrasos**

Dissertação de Mestrado  
Mestrado Integrado em Engenharia de Comunicações

Trabalho realizado sob orientação de  
**Doutor Joaquim Melo Henriques de Macedo**  
**Doutor António Luís Duarte Costa**

outubro de 2013

## **DECLARAÇÃO**

Nome: João Miguel Timóteo Antunes

Endereço eletrónico: joao99antunes@gmail.com

Telefone: 915359822

Número do Bilhete de Identidade: 13740687

Título da dissertação: Recolha e Análise de Dados de Contactos Físicos e Sociais numa Rede Tolerante a Atrasos

Orientadores: Doutor Joaquim Macedo e Doutor António Costa

Ano de conclusão: 2013

Designação do Mestrado: Mestrado Integrado em Engenharia de Comunicações

É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA TESE/TRABALHO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.

Universidade do Minho, \_\_\_/\_\_\_/\_\_\_\_\_

Assinatura: \_\_\_\_\_

## **AGRADECIMENTOS**

Agradeço aos meus orientadores, o Doutor António Luís Duarte Costa, e o Doutor Joaquim Melo Henriques de Macedo, pelos valiosos conselhos, motivação e confiança que me transmitiram ao longo deste processo.

A todos os meus amigos e colegas de curso, por todo o apoio, ajuda e alento que me deram nos piores momentos, principalmente ao Louis Costa, ao Bruno Lopes, ao Nuno Correia, ao António Lopes e ao Pedro Castro, que com a sua amizade me ajudaram a percorrer este trajeto académico.

Aos meus pais, Miguel e Emília, e à minha irmã Ana pelo apoio familiar constante e, principalmente, por me proporcionarem esta oportunidade.



# **Recolha e Análise de Dados de Contactos Físicos e Sociais numa Rede Tolerante a Atrasos**

## **RESUMO**

As redes tolerantes a atrasos surgiram com o propósito de abordar o problema de comunicação em redes onde a ligação é intermitente e feita através de contactos oportunistas. Um caso particular destas redes são aquelas em que os nós são dispositivos transportados por pessoas, as Pocket Switch Networks. Os primeiros estudos de protocolos de encaminhamento para estas redes baseavam-se sobretudo no conhecimento de encontros anteriores entre os nós.

Com o crescente uso das redes sociais e com a constante evolução nas tecnologias, aceder ao perfil social de uma pessoa tornou-se simples. Por isso, a exploração da afinidade social entre indivíduos, para melhorar a decisão de encaminhamento em Pocket Switch Networks está a ser recentemente alvo de estudo.

Neste trabalho, foi concebido um sistema de recolha de dados dos contactos físicos e sociais numa rede tolerante a atrasos com o objetivo de avaliar uma nova métrica social a ser usada na decisão de encaminhamento. Esta métrica relaciona a afinidade social e o histórico de contactos físicos entre os nós da rede, obtendo os dados sociais de uma rede social.

Resultados preliminares obtidos com dados de uma *dataset*, provam a validade da métrica apresentada, visto que os grafos sociais e físicos construídos possuem centralidades similares e correlacionadas.

**Palavras-chave:** Redes Tolerantes a Atrasos, Sistema de Recolha, Datasets, Pocket Switch Networks, Redes Sociais.



# **Data Collection and Analysis of Physical and Social Contacts in the Delay Tolerant Networks**

## **ABSTRACT**

The Delay Tolerant Networks have emerged with the purpose of addressing the problem of communication in networks, where the connection is intermittent and done through opportunistic contacts. A particular case of such networks is one in which the nodes are devices carried by people, the Pocket Switch Networks. Early studies of routing protocols for these networks were based mainly on the knowledge of previous encounters between nodes.

With the increasing use of social networks and the evolving technologies, access to the social profile of a person has become easy. Therefore, the operation of the social affinity between individuals to improve routing decision on Pocket Switch Networks is a new subject of study.

In this study, it was designed a system for collecting physical and social data in a Delay Tolerant Network to evaluate a new metric to be used in social forwarding decision. This metric relates the history of social affinity and physical contacts between the network nodes, collecting the social profile of the nodes in a social network.

Preliminary results with data from a dataset prove the validity of the displayed measure, because social and physical graphs constructed were correlated and have similar nodes centralities.

**Key-words:** Delay Tolerant Networks, Collection System, Datasets, Pocket Switch Networks, Social Networks.



## **ÍNDICE**

<b>DECLARAÇÃO</b>	<b>II</b>
<b>AGRADECIMENTOS</b>	<b>III</b>
<b>RESUMO</b>	<b>V</b>
<b>ABSTRACT</b>	<b>VII</b>
<b>ÍNDICE</b>	<b>IX</b>
<b>ÍNDICE DE FIGURAS</b>	<b>XII</b>
<b>ÍNDICE DE TABELAS</b>	<b>XIII</b>
<b>LISTA DE ACRÓNIMOS</b>	<b>XIV</b>
<b>CAPÍTULO I INTRODUÇÃO</b>	<b>1</b>
1.1 Objetivos	2
1.2 Contribuições	2
1.3 Organização do Documento	3
<b>CAPÍTULO II REDES TOLERANTES A ATRASOS</b>	<b>5</b>
2.1 Protocolo de Encaminhamento Epidémico	6
2.2 Prophet	8
2.3 SimBet	10
2.3.1 Centralidade	11
2.3.2 Similaridade	12
2.3.3 Força das Ligações	13
2.3.4 Encaminhamento	14
2.3.5 Conclusão	15
2.4 FairRouting	16
2.4.1 Encaminhamento	17
2.4.2 Força de Interação	17
2.4.3 Assortativeness	19

2.4.4 Conclusão	20
2.5 BubbleRap	20
2.5.1 Divisão de Comunidades	20
2.5.2 Centralidade nos Nós	22
2.5.3 Encaminhamento	22
2.5.4 Conclusão	24
2.6 PeopleRank	24
2.6.1 Encaminhamento	24
2.6.2 Conclusão	26
2.7 Análise de Desempenho	26
2.8 Bundle Protocol	29
2.8.1 Pacotes	31
2.8.2 Encaminhamento	32
<b>CAPÍTULO III SISTEMA DE RECOLHA E CONSTRUÇÃO DE GRAFOS</b>	<b>33</b>
3.1 Aplicação para a Recolha de Contactos Físicos e Sociais	34
3.1.1 Escolha da Aplicação	34
3.1.2 Comunicação entre os Nós da Rede	37
3.1.3 Obtenção do Perfil Social	41
3.2 Servidor para Armazenamento dos Dados	44
3.2.1 Registo do Histórico de Interações	45
3.3 Análise dos Dados	47
<b>CAPÍTULO IV IMPLEMENTAÇÃO</b>	<b>51</b>
4.1 Aplicação SocialConnector	51
4.1.1 Gestor de Perfil e Ligação ao Facebook	52
4.1.2 Comunicação entre Nós	54
4.1.3 Recolha e Armazenamento de Dados	58
4.1.4 Segurança e Privacidade	59
4.1.5 Comunicação com o Servidor	60
4.2 Servidor	61
<b>CAPÍTULO V TESTES E RESULTADOS</b>	<b>65</b>

5.1 O Dataset Sigcomm09	66
5.1.1 Preparação e Carregamento dos Dados	69
5.2 Grafo Histórico Contactos Físicos	71
5.3 Grafo Social	72
5.4 Grafo Coeficiente Físico	74
5.4.1 Comparação com Grafo Físico	76
5.5 Grafo Coeficiente Social	78
5.5.1 Comparação com Grafo Social	80
5.6 Cálculo Amizade entre os Nós	82
<b>CAPÍTULO VI CONCLUSÕES E TRABALHO FUTURO</b>	<b>85</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>87</b>

## ÍNDICE DE FIGURAS

Figura II.1 - Funcionamento Protocolo de Encaminhamento Epidémico baseado em [10]	xiii
Figura II.2 - Troca de mensagens Encaminhamento Epidémico	7
Figura II.3 - BubbleRap baseado em [14]	23
Figura II.4 - Algoritmo PeopleRank retirado de [5]	26
Figura II.5 - Posição do BP na camada OSI	29
Figura III.1 - Esquema Geral do Sistema de Recolha	33
Figura III.2 - Esquema SocialConnector	37
Figura III.3 - Bluetooth	41
Figura III.4 - Esquema Servidor	45
Figura IV.1 - Diagrama Obtenção Perfil Social	52
Figura IV.2 - Perfil do Utilizador	53
Figura IV.3 - Encontro entre Utilizadores	55
Figura IV.4 - Protocolo Comunicação entre os Nós	56
Figura IV.5 - Troca Lista Perfis	57
Figura IV.6 - Hashing MD5	59
Figura IV.7 - Ligação Aplicação-Servidor	61
Figura IV.8 - Ligação Grafo Servidor	62
Figura V.1 - Obtenção Contactos	70
Figura V.2 - Normalização Contactos	70
Figura V.3 - Grafo Físico	71
Figura V.4 - Distribuição Grau dos Nós Grafo Físico	72
Figura V.5 - Grafo Social	73
Figura V.6 - Distribuição Grau dos Nós Grafo Social	74
Figura V.7 - Grafo Coeficiente Físico	75
Figura V.8 - Distribuição Grau Dos Nós Grafo Coeficiente Físico	76
Figura V.9 - Grafo Coeficiente Social	79
Figura V.10 - Distribuição Grau dos Nós Grafo Coeficiente Social	79
Figura V.11 - Distribuição Amizade	83

## ÍNDICE DE TABELAS

Tabela II.1 - Comparação Protocolos	28
Tabela III.1 - Módulos SocialConnector	36
Tabela III.2 - Comparação Tecnologias de Comunicação	38
Tabela III.3 - Comparação Redes Sociais	43
Tabela III.4 - Módulos Servidor	44
Tabela IV.1 - Menu Inicial	51
Tabela IV.2 - Modos Aplicação	54
Tabela V.1 - Dataset Sigcomm09	68
Tabela V.2 - Comparação Centralidades dos Nós Grafo Histórico Contactos Físico e Grafo Coeficiente Físico	77
Tabela V.3 - Correlação Valor Centralidade Grafo Coeficiente Físico e Grafo Histórico de Contactos Físicos	78
Tabela V.4 - Correlação Valor Centralidade Grafo Social e Grafo Coeficiente Social	80
Tabela V.5 - Comparação Centralidade dos Nós Grafo Social e grafo Coeficiente Social	81

## LISTA DE ACRÓNIMOS

<b>Nome</b>	<b>Significado</b>
ACL	Assynchronous Connection-less
BP	Bundle Protocol
DTNRG	Delay Tolerant Networking Research Group
GSM	Global System for Mobile Communications
IRTF	Internet Research Task Force
NoSQL	Not only SQL
PSN	Pocket Switch Networks
RTA	Redes Tolerantes a Atrasos
SCO	Synchronous Connection Oriented
SDNV	Self-Delimiting Numeric Values
TCP/IP	Transmission Control Protocol/ Internet Protocol
UMTS	Universal Mobile Telecommunication System
WiFi	Wireless Fidelity
WNA	Weighted Network Analysis

# CAPÍTULO I INTRODUÇÃO

O enorme sucesso da Internet nas últimas três décadas deve-se ao uso de protocolos denominados TCP/IP, pois estes garantem a flexibilidade, eficiência e robustez, que lhe permitem suportar diversas aplicações em diferentes cenários. No entanto, em cenários com longos atrasos e conexões intermitentes, os protocolos TCP/IP não funcionam e é necessária a criação de novos protocolos. As redes com estas características específicas são denominadas Redes Tolerantes a Atrasos (RTAs), e um dos seus principais desafios é o encaminhamento, pois é necessário determinar rotas sem o conhecimento da existência de um caminho fim-a-fim[1].

Para contornar os problemas de longos atrasos e conexões intermitentes, as RTAs usam técnicas de troca de mensagens e armazenamento de dados. Quando uma mensagem precisa de ser enviada, ela é armazenada e encaminhada nó a nó desde a origem até o destino, ou seja, são redes do tipo *store-and-forward*. A mensagem é recebida integralmente e armazenada, para depois ser enviada ao próximo nó, que pode ou não ser o nó destino[2].

Um dos tipos de RTAs mais usuais são as chamadas *Pocket Switched Networks* (PSNs), que consistem em contactos oportunistas entre os nós da rede. Estes nós encontram-se em constante movimento e nem sempre estão conectados à rede, visto serem controlados diretamente por seres humanos, através de *smartphones*, computadores portáteis, etc.[3]. Este tipo de redes tira partido da mobilidade dos nós para a comunicação[4].

Tendo em conta que os nós das redes são controlados por humanos, os protocolos de encaminhamento para este tipo de redes podem basear-se em dois campos: o campo físico, que consiste no histórico de contactos<sup>1</sup> entre os nós da rede; e o campo social que consiste na relação social entre os mesmos. Existem muitos estudos relativos à relação entre estes dois campos [5] [6] [7], mas com este trabalho pretende-se igualmente verificar a validade de incluir uma métrica na decisão de encaminhamento, que relacione o histórico de contactos com a relação social entre os nós da rede, recorrendo a um perfil social a ser obtido através de uma rede social online.

---

<sup>1</sup> Um contacto é o encontro entre dois nós da rede, ou seja, quando um nó se encontra ao alcance de outro nó de modo a que estes possam comunicar

Para obter a relação social entre os nós, podemos utilizar as redes sociais existentes na Internet, como o Facebook, o Twitter ou o MySpace. Estas redes são utilizadas regularmente por milhões de pessoas em todo o mundo e permitem obter dados da vida social de cada um.

## **1.1 Objetivos**

Este trabalho de investigação propõe um sistema baseado no conceito Cliente-Servidor para a criação de uma RTA, através da qual se pretende obter os dados do histórico de contactos entre os nós e a relação social entre eles, de forma a proceder à posterior construção e análise de grafos[8] que permitam mostrar a relação entre os dados. Para tal, o trabalho está dividido em três partes distintas: uma aplicação em Android instalada em cada nó da RTA, responsável pela obtenção do histórico dos contactos físicos entre os nós e de um perfil social de cada nó enviando a informação obtida para o servidor; um servidor responsável por receber dados dos nós da rede e pela construção de um conjunto de grafos que relacione o histórico de contactos entre os nós e a relação social entre eles; um sistema de análise de grafos construído no servidor que permita obter resultados acerca do relacionamento entre o histórico de contactos e os dados sociais entre os nós numa RTA. Existem atualmente alguns estudos sobre a relação entre o grafo social e o grafo de contactos físicos [6] [5] [9], no entanto, o propósito final desta tese científica é verificar a validade da inclusão na decisão de encaminhamento de uma métrica que relaciona o campo social e o histórico de contactos físicos.

## **1.2 Contribuições**

Como resultado do trabalho realizado, obteve-se uma aplicação para Android, SocialConnector, que recolhe os contactos físicos e o perfil social de cada nó. Do servidor, que ao momento da escrita não está ainda terminado, é descrita a sua arquitetura e é simulado, sem a comunicação com os clientes, ou seja, unicamente lendo os dados de um *dataset* e construindo grafos. Foram também definidas duas métricas para calcular a afinidade social e relação física entre os nós, sendo provada a sua validade com os dados de um *dataset* que recolhe dados acerca os contactos entre os nós e o seu perfil social, o *dataset* sigcomm2009. Foi também publicado e apresentado um artigo sobre a tese desenvolvida na Conferência em

Redes de Computadores 2013 com o título “*Recolha e Análise de Dados de Contactos Físicos e Sociais numa Rede Tolerante a Atrasos*”.

### **1.3 Organização do Documento**

O documento está organizado da seguinte forma: no CAPÍTULO I é introduzido o problema, onde se expõe os objetivos do trabalho e referencia os resultados obtidos; o CAPÍTULO II consiste na análise e comparação de um conjunto de protocolos de encaminhamento para RTAs e no estudo do protocolo usado na camada de transporte em RTAs; no CAPÍTULO III é explicado em que consiste o trabalho desenvolvido, explicando a arquitetura do sistema, indicando quais os requisitos e funcionalidades do mesmo e é apresentada uma métrica para relacionar a relação física e social entre os nós numa RTA; no CAPÍTULO IV é explicado como foi implementado o sistema; o CAPÍTULO V usa um *dataset* pré-construído para verificar a validade das métricas propostas; por fim, no CAPÍTULO VI , concluímos o documento com algumas observações sobre o trabalho desenvolvido, indicando o trabalho futuro a ser realizado e os resultados obtidos.



## **CAPÍTULO II REDES TOLERANTES A ATRASOS**

Podemos dividir os protocolos de encaminhamento em RTAs em dois tipos distintos: protocolos baseados em redes onde existem infraestruturas e protocolos baseados em redes puramente móveis. Este último tipo de redes pode também ser dividido em dois tipos: protocolos baseados unicamente na disseminação e protocolos que têm informação de contexto na decisão de encaminhamento. A diferença entre estes dois tipos consiste na falta de informação de contexto no encaminhamento de mensagens entre os nós da rede nos protocolos de disseminação, enquanto que os protocolos baseados no contexto baseiam a decisão de encaminhamento de dados numa informação adicional, que pode incluir dados relativos ao histórico de contactos entre os nós ou da relação social entre eles, ou mesmo uma junção entre estes dois campos.

Nos protocolos de disseminação podemos incluir o protocolo de encaminhamento epidémico[10]. Este protocolo baseia-se na distribuição das mensagens pela rede de uma forma epidémica, isto é, a cada novo contacto entre dois nós da rede existe uma troca de informações entre os nós de modo a que cada nó receba as mensagens que ainda não tenha guardado. Desta forma, as mensagens são rapidamente distribuídas entre duas porções de rede que estejam próximas uma da outra, contando, para isso, com a mobilidade dos nós. Este protocolo garante uma probabilidade elevada na entrega de mensagens, contudo, sobrecarrega a rede devido à falta de informação de contexto e consome demasiados recursos aos nós da rede. Para contrariar a sobrecarga da rede e dos nós, posteriores versões deste protocolo incluíram um número limite de saltos e de cópias das mensagens presentes na rede.

Dentro dos protocolos baseados no contexto, incluem-se o Prophet[11], o SimBet[12], o FairRouting[13], o BubbleRap[14] e o PeopleRank[15]. Cada um destes protocolos utiliza uma informação de contexto na decisão de encaminhamento de mensagens entre os nós da rede, sendo que a diferença entre eles está na natureza dessa informação. O Prophet baseia-se unicamente no histórico de contactos entre os nós e na transitividade na decisão de encaminhamento, isto é, se dois nós se encontrarem regularmente, têm maior probabilidade de entregar uma mensagem um ao outro do que dois nós que raramente se encontrem. O SimBet inclui, na sua informação de contexto, a centralidade do nó da rede, a similaridade dos nós e a força das ligações entre eles. A informação de contexto no FairRouting inclui a força de interação

e o grau de *assortativeness* (pessoas com interesses comuns tendem a encontrar-se mais frequentemente). Já o BubbleRap utiliza a divisão dos nós em comunidades e a centralidade de cada um deles na decisão de encaminhamento. Por fim, o PeopleRank utiliza a relação social entre os nós da rede como informação de contexto.

Em relação ao protocolo de comunicação com os outros nós, existe um consenso em torno de um protocolo experimental que é usado na troca de mensagens numa rede RTA - o Bundle Protocol RFC 5050[16], que pode ser usado como suporte em todos os protocolos de encaminhamento. Este protocolo foi elaborado pelo IRTF's Delay Tolerant Networking Research Group (DTNRG), e divide blocos de dados em pacotes, transmitindo-os através de um mecanismo de *store-and-forward*.

## 2.1 Protocolo de Encaminhamento Epidémico

Este protocolo é um dos mais básicos de entre os que foram criados até ao momento e faz parte dos protocolos de disseminação, visto não ter em conta os contactos anteriores dos nós da rede, nem a relação social entre eles. A abordagem deste protocolo consiste na distribuição das mensagens pela rede como se de uma doença epidémica se tratasse. Desta forma, as mensagens são rapidamente distribuídas entre duas porções de rede que estejam próximas uma da outra, contando, para isso, com a mobilidade dos nós. Através desta forma de transmissão de mensagens, pode-se concluir que existe uma probabilidade muito elevada da mensagem ser entregue. Na Figura II.1, pode-se ver como se baseia o protocolo de encaminhamento epidémico.

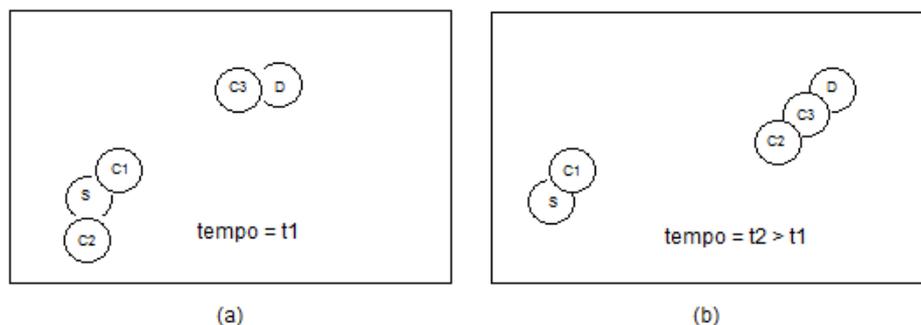


FIGURA II.1 - FUNCIONAMENTO PROTOCOLO DE ENCAMINHAMENTO EPIDÉMICO BASEADO EM [10]

Na Figura II.1 (a), um emissor S deseja enviar uma mensagem para D, mas não está diretamente ligado a ele, por isso, primeiro passa a mensagem para os seus vizinhos C1 e C2. Passado algum tempo, C2 entra em contacto com outro dispositivo C3 e passa a mensagem para este. C3 encontra-se próximo do destino da mensagem D entregando-lhe a mensagem de S (Figura II.1 (b)).

O protocolo de encaminhamento epidémico suporta a entrega de mensagens para destinos arbitrários, sem ter em consideração a topologia da rede e a conectividade entre os nós da mesma. Este protocolo baseia-se na distribuição transitiva de mensagens em redes *ad hoc*, com as mensagens a chegarem ao seu destino. Cada nó da rede mantém guardado um *buffer* com as mensagens criadas por ele e com as mensagens recebidas dos nós do qual entra em contacto.

Os nós organizam a lista de mensagens numa tabela e guardam um vetor que indica quais as mensagens que estão na tabela e o seu índice, o chamado *vetor sumário*. Quando dois nós da rede se encontram, o nó com o identificador menor inicia uma sessão com o outro nó. Para evitar redundância, cada nó guarda os identificadores dos nós com os quais falou recentemente. Durante a sessão anti isotrópica, os dois nós trocam entre si os respetivos *vetores sumários* para determinar quais as mensagens que estão em falta. Posteriormente, cada nó pede ao outro uma cópia das mensagens que lhe faltam, sendo que cada nó decide se aceita ou não a receção de uma determinada mensagem. Cada nó pode indicar um número máximo de mensagens que está disponível para encaminhar por outros, evitando, assim, um sobre carregamento da sua memória interna. Na Figura II.2, podemos observar como se efetua a troca de mensagens entre dois nós no protocolo de encaminhamento epidémico.



FIGURA II.2 - TROCA DE MENSAGENS ENCAMINHAMENTO EPIDÉMICO

Na Figura II.2, podemos observar que o nó A entra em contacto com o nó B e inicia a sessão anti isotrópica. Inicialmente, o A envia o seu *vetor sumário* ( $SV_A$ ) para o nó B (passo 1). De seguida, o B efetua um AND lógico entre o  $SV_A$  e a negação do seu próprio *vetor sumário* ( $\overline{SV_B}$ , que representa as mensagens que o B não possui). Assim, B determina a diferença entre as mensagens que o A possui e as suas próprias mensagens. Seguidamente, o B manda uma mensagem ao A a pedir as mensagens em falta (passo 2). Por fim, o A envia para o B as mensagens requisitadas por este (passo 3). Este processo é repetido sempre que um novo contacto entre nós se verifique. Se for proporcionado um buffer grande o suficiente, estas sessões anti isotrópicas acabam por garantir num determinado espaço de tempo a entrega de mensagens ao destino.

Depois de explicado este protocolo, podemos concluir que, apesar de este garantir a entrega total das mensagens ao recetor, requer, por seu turno, muitos recursos aos nós da rede (por exemplo memória interna e consumo de energia), e causa elevado tráfego da rede (visto que cada novo contacto entre os nós da rede implica uma troca de mensagens). Foram criados, portanto, protocolos que têm por base o encaminhamento das mensagens consoante os contactos físicos e o relacionamento social entre os nós da rede.

## 2.2 Prophet

Ao contrário do protocolo de encaminhamento epidémico e de outros protocolos, onde o encaminhamento é totalmente aleatório, na vida real os utilizadores normalmente não se movem de forma aleatória, movem-se de uma forma que pode ser prevista, baseada na repetição de padrões de comportamento, de modo a que se um nó visitar o mesmo local repetidas vezes seja bastante provável que o volte a visitar no futuro. Para fazer uso destas informações no melhoramento do encaminhamento de dados em RTAs, foi criado o protocolo Prophet, um protocolo com encaminhamento probabilístico que usa a história de contactos entre os nós e a transitividade na decisão de encaminhamento.

Para isso, o Prophet estabelece uma métrica denominada de *probabilidade de entrega*,  $P_{(a,b)} \in [0,1]$ , como a probabilidade do nó  $a$  em entregar a mensagem a um nó desconhecido  $b$ . A operação de troca de mensagens entre os nós no Prophet é parecida com o protocolo de encaminhamento epidémico. Isto é, quando dois nós se encontram, trocam um

*vetor sumário* que, neste caso, inclui ainda a informação da *probabilidade de entrega* guardada nos nós. Esta informação é usada para atualizar o vetor de *probabilidade de entrega* interno de cada nó e para este decidir que mensagens pedir ao outro nó, baseado na estratégia de encaminhamento detalhada a seguir.

O cálculo da *probabilidade de entrega* divide-se em três partes. A primeira, consiste na atualização da métrica aquando de um encontro com outro nó, de modo a que os nós que se encontrem frequentemente tenham uma elevada *probabilidade de entrega*. Este cálculo é demonstrado na Equação 1, onde  $P_{init} \in [0,1]$  é uma constante de inicialização.

$$P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) \times P_{init} \quad (1)$$

Se um par de nós não se encontrar durante um tempo, é menos provável que esses nós sejam bons encaminhadores de mensagens entre si, por isso, o valor da *probabilidade de entrega* deve decrescer com o tempo. A Equação 2 demonstra o cálculo dessa redução temporal, onde  $\gamma \in [0,1]$  é a constante de envelhecimento e  $k$  é o número de unidades de tempo que passaram desde a última vez que a métrica foi atualizada. A unidade de tempo pode diferir e ser definida com base na aplicação e nos atrasos previstos na rede.

$$P_{(a,b)} = P_{(a,b)old} \times \gamma^k \quad (2)$$

A *probabilidade de entrega* também tem uma propriedade transitiva, baseada na seguinte observação - se o nó A se encontra frequentemente com o nó B, e o nó B se encontra frequentemente com o nó C, então o nó C é provavelmente um bom nó para encaminhar mensagens para o nó A. A Equação 3 mostra como a transitividade afeta a *probabilidade de entrega*, onde  $\beta \in [0,1]$  é uma constante de escalabilidade que decide o impacto que a transitividade tem sobre a *probabilidade de entrega*.

$$P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) \times P_{(a,b)} \times P_{(b,c)} \times \beta \quad (3)$$

Numa rede RTA, quando uma mensagem chega a um nó, pode não haver um caminho para o destino da mensagem, e, portanto, o nó deve guardar a mensagem, e a cada novo encontro com um nó deve tomar a decisão de transferir ou não a mensagem para esse nó. Também é possível que a mensagem seja enviada para vários nós de modo a aumentar a probabilidade de uma mensagem ser entregue.

Estas decisões de encaminhamento não são fáceis de tomar. Deve haver sensibilidade na escolha de alguns parâmetros. Por exemplo, em alguns casos deve ser utilizado um valor fixo de *threshold*, e só quando um nó tiver uma *probabilidade de entrega* acima desse valor é que deve receber mensagens para encaminhar. Existe, igualmente, a questão de decidir a quantos nós devemos passar a mensagem, sendo que um maior número de nós implica maiores recursos do sistema, mas aumenta a probabilidade de entrega da mensagem; enquanto que um menor número de nós poupa recursos ao sistema, mas diminui essa probabilidade.

No Prophet, definiu-se que uma mensagem só é passada de um nó para outro se o nó que tem a mensagem tiver um menor valor de *probabilidade de entrega* do que o nó encontrado.

Pode concluir-se que o Prophet é um protocolo mais sofisticado que o protocolo de encaminhamento epidémico, pois tem em conta o histórico de contactos entre os nós e a transitividade. Para além disso, em termos de eficiência, supera o protocolo de encaminhamento epidémico, pois para além de garantir uma probabilidade de entrega de mensagens parecida com este, diminui significativamente o tráfego na rede.

### 2.3 SimBet

O protocolo SimBet utiliza, na decisão de encaminhamento, dados sociais e de contactos físicos dos nós numa RTA. A decisão de encaminhamento neste protocolo tem em conta três componentes diferentes: a centralidade de cada nó, a força das ligações entre os nós e a previsão dessas mesmas ligações (i.e. similaridade entre os nós).

### 2.3.1 Centralidade

A centralidade de um nó numa rede é a quantificação da importância relativa desse nó na rede (por exemplo, mede o quanto uma pessoa é importante numa rede social). Normalmente, um nó central tem um maior valor de centralidade, pois tem uma maior capacidade de se ligar aos outros nós da rede. Existem várias maneiras de calcular a centralidade de um nó numa rede, mas a forma utilizada neste algoritmo é chamada de centralidade *betweenness*[17].

A centralidade *betweenness* pode ser encarada como uma medida do controlo que um nó tem sobre a informação que é trocada entre os outros nós. Um nó com um valor mais elevado de centralidade *betweenness* tem uma maior capacidade de facilitar a comunicação entre os nós vizinhos.

No protocolo SimBet a centralidade *betweenness* é calculada usando uma rede ego - i.e. uma rede que inclui o próprio nó (nó ego) e os nós a que este está ligado, bem como as ligações entre eles - para representar os nós com que o nó entrou em contacto. Quando dois nós se encontram, trocam uma lista de contactos. Um contacto é definido como o encontro de um nó com outro. A lista de contactos recebida é usada para atualizar a rede ego de cada nó. Matematicamente representa-se a lista de contactos por uma matriz adjacente  $A$ , que é uma matriz simétrica  $n \times n$ , onde  $n$  é o número de contactos com outros nós que o nó tem. A matriz  $A$  tem como elementos:

$$A_{i,j} = \begin{cases} 1, & \text{se houver um contato entre } i \text{ e } j \\ 0, & \text{caso contrário} \end{cases} \quad (4)$$

A centralidade *betweenness* é calculada computando os nós que estão indiretamente ligados através do nó ego. A centralidade *betweenness* do nó ego é o somatório dos inversos das entradas da matriz  $A'_{i,j} = A^2[1 - A]_{i,j}$  [18]. O valor da centralidade *betweenness* é obtido através da Equação 5.

$$\sum \frac{1}{A'_{i,j}} \quad (5)$$

### 2.3.2 Similaridade

Como iremos ver em 2.3.3 a força das ligações mede a força das conexões existentes entre os nós da rede. Por outro lado, existem previsores que usam informação do passado dos nós para prever ligações que podem acontecer futuramente. A probabilidade de dois nós se encontrarem é mais elevada se eles tiverem nós conhecidos em comum. Este fenómeno é chamado de *clustering*, i.e., a probabilidade de dois nós se encontrarem é mais elevada se esses nós tiverem vizinhos em comum.

Uma das formas de prever uma possível ligação é medir a similaridade entre dois nós, ou seja, medir os vizinhos em comum entre os nós. No protocolo SimBet, a similaridade é calculada usando a mesma matriz  $n \times n$  discutida em 2.3.1. O número de vizinhos em comum entre dois nós pode ser obtido através da sobreposição da sua lista de contactos, ou seja, da sua matriz  $n \times n$ .

Assim, apenas podemos obter a similaridade para nós que estão ligados diretamente, por isso, os nós guardam separadamente uma matriz  $n \times m$  com uma lista de contactos que podem ser indiretamente obtidos através de nós vizinhos, sendo  $n$  o número de nós ligados diretamente e  $m$  o número de nós que ainda não foram encontrados mas que podem ser encontrados indiretamente. O cálculo da similaridade é obtido da forma descrita na Equação 6.

$$Sim(n, e) = |N_n \cap N_e| \quad (6)$$

Onde  $N_n$  e  $N_e$  são o conjunto de contactos mantidos pelos nós  $n$  e  $e$  respetivamente.

### 2.3.3 Força das Ligações

Para medir a força de uma ligação deve considerar-se vários indicadores acerca da ligação, no caso deste protocolo, usaram-se os seguintes indicadores: a frequência com que um nó se encontra com outro, a intimidade/proximidade entre os nós e quão recente foi a última interação entre os nós.

O indicador de frequência de encontros entre os nós  $n$  e  $m$  é calculado da seguinte forma:

$$FI_n(m) = \frac{f(m)}{F(n) - f(m)} \quad (7)$$

Onde:

- $f(m)$  é o número de vezes que o nó  $n$  encontrou o nó  $m$ ;
- $F(n)$  é o número total de encontros do nó  $n$  na rede.

O indicador de intimidade/proximidade está relacionado com a duração dos contactos entre os nós e pode ser calculado da seguinte forma:

$$ICI_n(m) = \frac{d(m)}{D(n) - d(m)} \quad (8)$$

Onde:

- $d(m)$  é a quantidade de tempo total que o nó  $n$  esteve ligado ao nó  $m$ ;
- $D(n)$  é a quantidade de tempo total que o nó  $n$  esteve ligado com todos os nós na rede.

O indicador de quão recente foi o contacto entre os nós é calculado da seguinte forma:

$$RecI_n(m) = \frac{rec(m)}{L(n) - rec(m)} \quad (9)$$

Onde:

- $rec(m)$  é o tempo do contacto mais recente entre o nó  $n$  e o nó  $m$ ;
- $L(n)$  é o tempo total que o nó  $n$  faz parte da rede.

Por fim, a força da ligação entre o nó  $n$  e o nó  $m$  é uma combinação dos três indicadores. Uma ligação para ser considerada forte deve ter um valor elevado nos três indicadores.

Consequentemente, a força de ligação de um nó  $n$  para um nó  $m$  é dada pela Equação 10.

$$TieStrength_n(m) = \sum_{I \in T} I_n(m) \quad (10)$$

Onde  $T$  representa o conjunto de indicadores sociais considerados.

### 2.3.4 Encaminhamento

O SimBet usa a centralidade no encaminhamento, pois esta mostra ser bastante útil para encontrar caminhos em redes estáticas. No entanto, o uso deste indicador provoca a congestão de dados nos nós mais importantes e não tem em conta a variação das ligações no tempo. Com a medição da força das ligações, podemos ultrapassar este problema, identificando as ligações que tem maior probabilidade de estar disponíveis. Ainda assim, este indicador não tem em conta a evolução dinâmica da rede ao longo do tempo. O uso de previsores de ligação é usado, então, para prever futuras ligações na rede. O protocolo SimBet propõe, portanto, uma combinação da centralidade com a medição da força das ligações e com os previsores de ligações futuras para definir a decisão de encaminhamento numa rede com uma estrutura social associada.

O encaminhamento é feito da seguinte forma: quando dois nós se encontram, trocam uma lista de nós com os quais se encontraram previamente. Essa lista é usada para calcular localmente três utilidades: *betweenness*, similaridade e força das ligações. Posteriormente, cada nó calcula uma utilidade, chamada de SimBetTS, para o nó destino de cada mensagem que tem para encaminhar. A mensagem é então encaminhada para o nó com uma utilidade SimBetTS mais elevada para com o nó destino da mensagem.

As utilidades *betweenness*  $BetUtil_n$ , similaridade  $SimUtil_n$  e força dos nós  $TSUtil_n$  do nó  $n$  para entregar uma mensagem ao nó  $d$  comparando com o nó  $m$  são calculadas da seguinte forma:

$$SimUtil_n(d) = \frac{Sim_n(d)}{Sim_n(d) + Sim_m(d)} \quad (11)$$

$$BetUtil_n(d) = \frac{Bet_n}{Bet_n + Bet_m} \quad (12)$$

$$TSUtil_n(d) = \frac{TieStrength_n(d)}{TieStrength_n(d) + TieStrength_m(d)} \quad (13)$$

Por fim é preciso calcular a utilidade SimBetTS que irá servir para o encaminhamento:

$$SimBetTSUtil_n = \sum_{u \in U} u_{n(d)} \quad (14)$$

Sendo  $U = \{SimUtil, BetUtil, TSUtil\}$ .

### 2.3.5 Conclusão

Com este protocolo foram introduzidas novas técnicas de análise social e foi demonstrado como elas podem interferir no encaminhamento de uma mensagem numa rede RTA. O protocolo combina três métricas: centralidade *betweenness*, similaridade e força dos nós. A combinação

destas três métricas resulta numa melhoria na entrega de mensagens, com a vantagem da redução de carga nos nós centrais, originando uma melhor distribuição de carga pela rede.

## 2.4 FairRouting

O encaminhamento em redes tolerantes a atrasos (RTAs) requiere a resposta a duas questões: como é que o próximo melhor nó é escolhido e como é que o tráfego é distribuído pelos utilizadores. A maior parte dos protocolos existentes focam-se na maximização do débito efetivo e na descoberta dos caminhos mais curtos para obter algoritmos eficientes com elevado rácio de mensagens entregues ao destino, mas salvo algumas exceções, não se deu muita atenção à escalabilidade e à confiabilidade.

As ligações usadas nos algoritmos de encaminhamento para RTAs estão relacionadas com redes sociais, que, por seu lado, são um caso particular de redes complexas. Uma das características destas redes é exibirem uma distribuição de conectividade do tipo *fat-tail*, onde poucos nós têm muitas conexões, enquanto que a maioria dos nós tem poucas. Uma vez que as mensagens são enviadas quando existe um encontro entre dois nós, é inevitável que os nós com mais ligações transportem a maioria do tráfego, produzindo assim, uma distribuição de carga injusta.

Infelizmente, a justiça na distribuição de carga pelos nós da rede só é possível se forem aplicados processos heurísticos nas decisões de encaminhamento. Em sistemas igualmente complexos como a Internet, as redes de estradas ou o tráfego aéreo, o problema da injustiça é resolvido aumentando os recursos dos nós onde passa mais tráfego, por exemplo, instalação de *switches* mais rápidos, a construção de mais autoestradas ou a construção de mais terminais num aeroporto. Contudo, esta solução não pode ser implementada nas RTAs, pois, nestas redes, cada nó pertence a um diferente domínio administrativo (i.e. utilizadores individuais).

Devido a estas circunstâncias, falhas podem ocorrer devido a várias razões: os nós não têm recursos suficientes para lidarem com a carga e falham; os nós importantes são logicamente os alvos preferenciais para ataques; e a relação entre o custo total (total de mensagens a enviar) e a utilidade (mensagens próprias a serem enviadas) nos nós com mais ligações é um impedimento para a participação destes na rede, sendo estes nós, ironicamente, os mais importantes na rede. Estes problemas ilustram a necessidade de uma distribuição de

carga justa nos algoritmos de encaminhamento para RTAs. O protocolo FairRoute tenta dar uma solução para resolver esse problema.

### 2.4.1 Encaminhamento

Nas RTAs, as mensagens são encaminhadas de nó em nó quando há contacto entre os nós. Devido ao traço de contactos ser uma rede social, passamos para as ciências sociais para desenhar um algoritmo de encaminhamento que ultrapasse o problema de injustiça dos protocolos já existentes. Particularmente, o protocolo FairRoute baseia-se na *força de interação* e na *assortativity* [13] para a decisão de encaminhamento quando existe um contacto entre dois nós. Estas duas medidas irão ser descritas de seguida.

### 2.4.2 Força de Interação

A força de interação representa a avaliação subjetiva da força de uma relação entre dois indivíduos, e pode ser usada como um indicador da probabilidade de um contacto ser sustentado ao longo do tempo. O protocolo FairRoute usa dois diferentes estimadores para a força de interação que operam em diferentes escalas de tempo:  $\sigma_{ij}$  que indica a força de interação entre  $i$  e  $j$  a curto prazo; e  $\lambda_{ij}$  que indica a força de interação numa escala de tempo a longo prazo. A força de interação entre  $i$  e  $j$  aumenta em cada contacto, mas diminui ao longo do tempo a uma taxa exponencial de  $r_\sigma$  e  $r_\lambda$  para a força de interação a curto e longo prazo, respetivamente. Assim, é requerido que  $r_\lambda \ll r_\sigma$ .

Quando um contacto entre  $i$  e  $j$  acontece, os nós atualizam a sua perceção da força de interação da seguinte forma:

$$\sigma_{ik} = \sigma_{ik} e^{-r_\sigma (t-t_i)} \quad \forall k \in N_i, \quad (15)$$

$$\lambda_{ik} = \lambda_{ik} e^{-r_\lambda (t-t_i)} \quad \forall k \in N_i, \quad (16)$$

$$(\sigma_{ij}, \lambda_{ij}) = (\sigma_{ij}, \lambda_{ij}) + (1, 1). \quad (17)$$

Onde  $N_i$  é a lista dos contactos do nó  $i$ ;  $t_i$  é o tempo do último contacto do nó  $i$  com outro nó qualquer da rede; e  $t$  é o tempo atual. A cada contacto, o nó  $i$ : atualiza a redução exponencial da força de interação ( $r_\sigma$  e  $r_\lambda$ ) com todos os nós que tenha encontrado no passado ( $N_i$ ); aumenta a força de interação com o nó  $j$  por 1 (a curto e longo prazo); e, por fim, atualiza o tempo do último contacto ( $t_i = t$ ).

De seguida, é definido o agregado da força de interação  $s_{ij}$  entre os nós  $i$  e  $j$  como  $s_{ij} = \lambda_{ij}(\lambda_{ij} - \sigma_{ij})$ . Intuitivamente, o agregado de força de interação é uma indicação da frequência de interações a longo termo (proporcional a  $\lambda_{ij}$ ), penalizando, assim, rajadas esporádicas de atividade (proporcional à diferença entre as escalas de longo e curto termo ( $\lambda_{ij} - \sigma_{ij}$ )). Definimos  $u_{ijk}$  como a percepção de  $i$  da utilidade do nó  $j$  para entregar uma mensagem ao nó  $k$  como:

$$u_{ijk} = \frac{\lambda_{jk}(\lambda_{jk} - \sigma_{jk})}{\lambda_{jk}(\lambda_{jk} - \sigma_{jk}) + \lambda_{ik}(\lambda_{ik} - \sigma_{ik})} \quad (18)$$

A Equação 18 representa a utilidade do nó  $j$  para entregar uma mensagem a  $k$  vista pelo nó  $i$ , normalizado pela utilidade total. Para valores  $u_{ijk} > 0.5$ , espera-se que o nó  $j$  seja melhor para entregar a mensagem a  $k$  que o nó  $i$ . A utilidade  $u_{ijk}$  é definida apenas quando  $\lambda_{ik} + \lambda_{jk} > 0$ , caso contrário, é zero.

De forma análoga, seja  $u_{ij}$  a percepção de  $i$  da utilidade de  $j$  em entregar uma mensagem a um nó qualquer:

$$u_{ij} = \frac{\sum_{k \in N_j} \lambda_{jk}(\lambda_{jk} - \sigma_{jk})}{\sum_{k \in N_j} \lambda_{jk}(\lambda_{jk} - \sigma_{jk}) + \sum_{k \in N_i} \lambda_{ik}(\lambda_{ik} - \sigma_{ik})} \quad (19)$$

Finalmente o nó  $i$  irá encaminhar a mensagem para  $j$  sendo o destino  $k$  se e só se:

$$\begin{cases} u_{ijk} > \frac{1}{2} \wedge (\lambda_{ik} + \lambda_{jk}) > 0 \\ u_{ij} > \frac{1}{2} \wedge (\lambda_{ik} + \lambda_{jk}) = 0 \end{cases} \quad (20)$$

A heurística da Equação 20, por si só, não permite obter uma distribuição de tráfego completamente equilibrada. A razão deve-se à ganância de certos utilizadores, com a decisão de encaminhamento inclinada a fazer passar a mensagem pelos nós com mais conectividade. Para contrariar este efeito, podemos virar-nos, de novo, para a sociologia e observar os mecanismos segundo os quais as pessoas decidem com quem interagem. Podemos dizer que pessoas com os mesmos gostos e interesses profissionais e pessoais tendem a interagir entre elas e a não interagir com pessoas de diferentes grupos sociais. Este comportamento, chamado de *assortativeness* [14], é um dos fatores determinantes da forma como as pessoas interagem umas com as outras.

### 2.4.3 Assortativeness

Para obter o valor da *assortiveness* no protocolo FairRoute, define-se o estatuto social de um nó  $i$  na RTA como equivalente ao tamanho da fila de mensagens no nó  $Q_i$ . Este tamanho da fila pode ser interpretado como um medidor da popularidade do nó. Uma vez que aceitar encaminhar uma mensagem tem o seu custo, os nós só vão aceitar encaminhar mensagens de nós com um estatuto social igual ou superior. Um nó  $i$  só encaminhará uma mensagem para  $k$  através de  $j$  se as seguintes condições se verificarem:

$$\begin{cases} \left( u_{ijk} > \frac{1}{2} \right) \wedge (\lambda_{ik} + \lambda_{jk}) > 0 \wedge (Q_j \leq Q_i) \\ (u_{ijk} = 1) \wedge (\lambda_{ik} + \lambda_{jk}) > 0 \\ \left( u_{ij} > \frac{1}{2} \right) \wedge (\lambda_{ik} + \lambda_{jk}) = 0 \wedge (Q_j \leq Q_i) \end{cases} \quad (21)$$

Assim sendo, os nós com um estatuto mais elevado são capazes de encaminhar as suas mensagens mais rapidamente que os restantes nós devido à sua posição privilegiada. Os nós

com um estatuto mais baixo têm de encontrar caminhos alternativos, sendo que, nas redes sociais e de contacto, existem diversos caminhos alternativos, evitando, assim, perda de *débito efetivo*.

#### **2.4.4 Conclusão**

A maior contribuição do protocolo FairRoute para o estado da arte é que, ao contrário de outros algoritmos baseados em encaminhamento social, distribui a carga pelos nós de uma forma justa, permitindo, portanto, obter mais eficiência sem perda de *débito efetivo*.

### **2.5 BubbleRap**

Este protocolo é incluído nos protocolos do tipo de protocolos baseados no contexto, pois tem em consideração alguns parâmetros sociais e físicos entre os nós da rede antes de passar uma mensagem para um encaminhador. O protocolo centra-se em duas métricas sociais: a divisão entre comunidades e a centralidade de cada nó.

Na sociedade, a cooperação liga as pessoas mas também as divide em diferentes comunidades, tendo em conta diferentes interesses e opiniões. O BubbleRap é um protocolo que usa um algoritmo híbrido para o encaminhamento, que seleciona os nós com maior centralidade e os divide em diferentes comunidades para servirem como encaminhadores para uma determinada mensagem. Como sabemos, numa PSN não temos informação *a priori* sobre a infraestrutura de rede e, portanto, a deteção e disseminação distribuída da popularidade e da comunidade de cada nó é fulcral para a escolha do melhor encaminhador para uma mensagem. Com este protocolo, podemos verificar que isso não só é possível, como tem grande eficiência na entrega de mensagens.

#### **2.5.1 Divisão de Comunidades**

Para a deteção de comunidades, o protocolo BubbleRap utiliza dois algoritmos que se complementam: o K-CLIQUE e o *weighted network analysis* (WNA). Para entender melhor estes

algoritmos, temos de introduzir o conceito de grafo de contactos. Este grafo é construído com base no número de contactos e a sua duração entre os nós da rede. Por exemplo, para medir a relação entre duas pessoas (i.e. dois nós) usamos o número de vezes que eles se encontraram e o tempo que eles estiveram em contacto. Naturalmente, duas pessoas que estejam mais tempo juntas ou que se encontrem variadas vezes, têm uma relação mais chegada.

O algoritmo K-clique foi introduzido por Palla e define uma comunidade k-clique como uma união de todos os k-cliques (subgrafos completos de tamanho k) que podem ser alcançáveis uns dos outros a partir de k-cliques adjacentes, onde dois k-cliques se dizem adjacentes se partilharem k-1 nós da rede. À medida que k aumenta, as comunidades k-cliques encolhem, mas, por outro lado, tornam-se mais coesas visto que cada nó é obrigado a fazer parte de, pelo menos, uma comunidade k-clique.

O algoritmo WNA de Newton divide a rede em comunidades, calculando o valor de modularidade (Q) de cada uma através da Equação 22.

$$Q = \sum_{vw} \left[ \frac{A_{vw}}{2m} - \frac{k_v k_w}{(2m)^2} \right] \delta(c_v, c_w) \quad (22)$$

Onde:

- $A_{vw}$  é o valor do peso da ligação entre o vértice v e w caso exista, sendo 0 se não existir;
- A função  $\delta(c_v, c_w)$  é 1 caso  $c_v = c_w$  e 0 caso contrário;
- $m = \frac{1}{2} \sum_{vw} A_{vw}$ ;
- $k_v$  é o grau do vértice v definido por  $\sum_w A_{vw}$ ;
- $c_v$  é a comunidade á qual o vértice v pertence.

A modularidade é entendida como a diferença entre esta fração e a fração das ligações que seriam esperadas nas comunidades, se as ligações entre os vértices fossem aleatórias. Este algoritmo é, basicamente, um algoritmo genérico que usa a modularidade como uma medida de adequação.

Os usos destes dois algoritmos de deteção de comunidades centralizados dão-nos informações valiosas para fazer o *clustering* social das pessoas, e são úteis para análise de dados em offline de ligações móveis previamente obtidas. Podemos usá-los para explorar estruturas nos dados para, posteriormente, desenhar melhores estratégias de encaminhamento e medidas de segurança.

### **2.5.2 Centralidade nos Nós**

Na sociedade humana, as pessoas têm diferentes níveis de popularidade, por isso, com este protocolo, pretende-se implementar heterogeneidade na popularidade para ajudar a desenhar políticas de encaminhamento mais eficientes: passar a mensagem para um encaminhador mais popular do que para um menos popular.

Uma rede temporal é um tipo de rede ponderada. A medida centralidade usada nas redes consideradas normais não funciona neste tipo de redes (i.e. estas redes são dinâmicas e as ligações são dependentes do tempo), assim, precisamos de uma forma diferente de calcular a centralidade de cada nó no sistema. A abordagem do BubbleRap é a seguinte:

1. Realizar um grande número de emulações de dados com diferentes padrões de tráfego uniformemente distribuídos, usando, para isso, o emulador HuggleSim.
2. Contar o número de vezes que um nó funciona como encaminhador para outros nós da rede quando é usado o caminho mais curto na entrega, isto quando a mesma mensagem é entregue ao destinatário por caminhos diferentes.

Ao número calculado usando as duas variantes acima descritas chamamos centralidade *betweenness* do nó no grafo temporal.

### **2.5.3 Encaminhamento**

O protocolo de encaminhamento definido no BubbleRap funciona da seguinte forma: se um nó tem uma mensagem destinada para outro nó, o nó emissor deve primeiro passar essa mensagem para cima na hierarquia (baseada no ranking global de popularidade/centralidade) até que atinja um nó da mesma comunidade do nó recetor da mensagem. De seguida, usando o sistema de ranking local, a mensagem é passada de nó em nó (*bubbling*) na comunidade até

chegar ao destino ou a validade da mensagem ser expirada. Este método não requer que um nó saiba o ranking dos outros nós todos do sistema, mas sim que seja possível comparar o ranking do nó que encontrar, e passar-lhe a mensagem. Este protocolo é chamado de Bubble, porque cada comunidade é como uma bolha. Na Figura II.3, podemos observar o funcionamento deste protocolo.

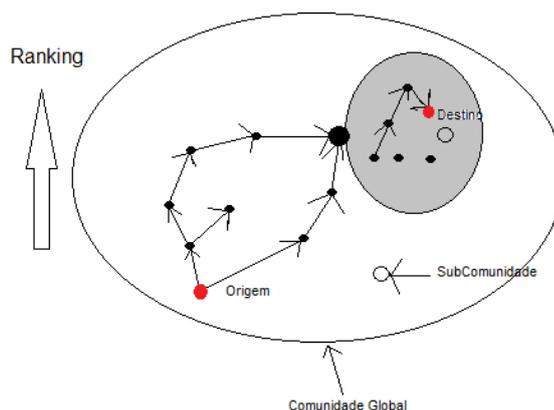


FIGURA II.3 - BUBBLERAP BASEADO EM [14]

O funcionamento deste protocolo obedece à nossa intuição do que é o mundo real. Primeiro, tenta encaminhar os dados para as pessoas mais populares à sua volta e, posteriormente, passá-los para pessoas bem conhecidas na sociedade, como por exemplo, um carteiro. Quando o carteiro se encontrar com um membro da mesma comunidade que o destinatário, a mensagem irá passar para essa comunidade. Esse membro da comunidade que receber a mensagem vai tentar procurar as pessoas mais populares dentro da comunidade e passar-lhes a mensagem até que ela chegue ao destino, ao membro mais popular ou expire.

Numa versão modificada deste protocolo, quando uma mensagem é entregue à comunidade destino, o encaminhador inicial pode apagar essa mensagem da sua memória para evitar uma futura disseminação, isto assumindo que o membro da comunidade que recebe a mensagem a vai conseguir entregar. Este protocolo modificado é chamado de Bubble-B, enquanto o original é chamado de Bubble-A.

### **2.5.4 Conclusão**

Neste protocolo, a divisão entre comunidades e a centralidade dos nós pode ser usada como medida social para encaminhamento de mensagens em PSNs. O algoritmo foi desenhado para RTAs, com os dispositivos a serem carregados por humanos, permitindo-nos dizer que a percentagem de entrega de mensagens é similar ao protocolo SimBet, mas com o uso de menos recursos no fluxo e no controlo de fluxo. O BubbleRap está desenhado para funcionar melhor num sistema em hierarquia.

## **2.6 PeopleRank**

O protocolo PeopleRank, tal como os protocolos previamente estudados, utiliza uma abordagem para o uso das interações sociais entre os nós numa RTA para a decisão de encaminhamento das mensagens. O algoritmo é oportunista e usa o conceito de PeopleRank para medir a “importância” de um nó no grafo social. Este protocolo é baseado no algoritmo PageRank usado no motor de pesquisa Google para medir a importância relativa de uma determinada página Web, dentro de um conjunto de páginas[19]. Analogamente, o PeopleRank identifica os nós mais populares num contexto social para serem os encaminhadores da mensagem, visto que os nós mais populares têm mais hipóteses de encontrar outros nós na rede.

### **2.6.1 Encaminhamento**

O uso dos dados da interação social entre as pessoas numa RTA no encaminhamento de dados já mostrou, como podemos observar nos protocolos já estudados, ser muito proveitoso para reduzir o número de réplicas de cada mensagem na rede, ao mesmo tempo que mantém a probabilidade de uma mensagem ser entregue.

Com a evolução do uso das redes sociais, tais como o Facebook, o Twitter ou o MySpace, a informação acerca das interações sociais entre os utilizadores ficou mais facilmente acessível. A ideia do protocolo PeopleRank é usar os dados sociais obtidos nestas redes, visto serem dados mais estáveis do que aqueles obtidos através dos contactos oportunistas entre os nós numa

RTA. O protocolo PeopleRank usa um algoritmo que mede a popularidade de cada nó (PeopleRank) e que usa essa medida na decisão de encaminhamento, por exemplo, se um nó  $v$  encaminha um mensagem para o nó  $u$ , isso significa que o nó  $u$  tem um melhor ranking do que  $v$ . A ideia geral é que um nó com um valor de PeopleRank maior, seja mais central na rede.

No PeopleRank, uma pessoa é definida como “importante” se esta está ligada, no contexto social, a muitas outras pessoas “importantes”. É estabelecido um grafo social entre as pessoas (nós) na rede quando estas estão relacionadas socialmente. Essas relações podem ser medidas através de amizades explícitas (redes sociais), comunicação pessoal (telemóveis) ou interesses em comum. Define-se, então, um grafo social  $G_s = (V_s, E_s)$ , sendo os vértices do grafo definidos por  $V_s$  e as ligações entre eles por  $E_s$ . Uma ligação  $(u, v) \in E_s$  só existe se houver uma relação social entre o nó  $u$  e o nó  $v$ . Neste protocolo, definimos que um nó  $v$  e um nó  $u$  têm uma relação social em dois casos: se são declaradamente amigos ou se partilham um número  $k$  de interesses comuns.

Assim sendo, o valor de PeopleRank é obtido através da Equação 23.

$$PeR(N_i) = (1 - d) + d \sum_{N_j \in F(N_i)} \frac{PeR(N_j)}{|F(N_j)|} \quad (23)$$

Onde:

- $N_i$  corresponde a um nó da rede;
- $F(N_i)$  é o conjunto de nós vizinhos de  $N_i$  ;
- $d$  é o fator de *damping*.

O valor de *damping* define-se como a probabilidade de que, a qualquer encontro entre nós, a relação social entre eles ajude a melhorar o seu ranking. Isto significa que, quanto maior for o valor de  $d$ , mais o algoritmo tem em conta a relação social entre os nós na decisão de encaminhamento.

Nas RTAs, as ligações estão em constante mutação, por isso, foi necessário definir um algoritmo para aplicar o conceito estudado até agora na decisão de encaminhamento. Assim, quando dois nós se encontram neste tipo de redes, trocam dois tipos de informação entre si: o

seu valor de PeopleRank atual e o número de grafos sociais vizinhos que têm. De seguida, os dois nós atualizam o valor do seu Peoplerank usando a fórmula previamente estudada. Na Figura II.4, podemos observar o algoritmo utilizando pseudo código:

---

```

Require:  $|F(i)| \geq 0$ 
 $PeR(i) \leftarrow 0$ 
while 1 do
  while  $i$  is in contact with  $j$  do
    if  $j \in F(i)$  then
       $send(PeR(i), |F(i)|)$ 
       $receive(PeR(j), |F(j)|)$ 
       $update(PeR(i))$ 
    end if
    while  $\exists m \in buffer(i)$  do
      if  $PeR(j) \geq PeR(i)$  OR  $j = destination(m)$  then
         $Forward(m, j)$ 
      end if
    end while
  end while
end while

```

(Eq. 2)

---

FIGURA II.4 - ALGORITMO PEOPLERANK RETIRADO DE [5]

Como podemos observar, neste algoritmo os nós atualizam o seu PeopleRank frequentemente. De facto, cada vez que dois nós se encontram, o seu Peoplerank aumenta.

## 2.6.2 Conclusão

Com o emergente crescimento do uso de redes sociais pelas pessoas, obter a informação social dos relacionamentos entre elas tornou-se mais fácil. O PeopleRank, utiliza um algoritmo que explora as propriedades sociais para reduzir o número de réplicas de cada mensagem em RTAs. O protocolo mede a importância de cada nó na rede, usando grafos sociais baseados nos contactos entre os nós da rede e a sua relação social.

## 2.7 Análise de Desempenho

Nesta secção, será realizada uma análise do desempenho de todos os protocolos detalhados anteriormente, recorrendo-se a uma comparação entre os seguintes aspetos: a percentagem das mensagens entregues, o atraso das mensagens, o congestionamento na rede

e outros parâmetros específicos de cada protocolo. Os protocolos estudados incluem o protocolo de encaminhamento epidémico, o Prophet, o BubbleRap, o FairRouting, o SimBet e o PeopleRank.

Em relação ao protocolo de encaminhamento epidémico, podemos dizer que este protocolo garante uma elevada percentagem na entrega de mensagens aos destinatários como era previsto, no entanto, para isso, requer demasiados recursos aos dispositivos da rede, como por exemplo, ter um poder de alcance elevado, ter elevada capacidade de memória interna, entre outras coisas. Outra característica deste protocolo é fazer variar o tempo de entrega de mensagens (atraso) e a percentagem de mensagens entregues consoante: o número limite de saltos das mensagens, do alcance de cada dispositivo (potência de transmissão) e da capacidade de memória dos buffers dos dispositivos. Assim, podemos considerar que este torna-se um protocolo razoável quando não se está preocupado com o congestionamento da rede, nem tem de haver preocupações quanto aos recursos disponíveis em cada nó da rede, o que, na atualidade, raramente acontece, fazendo com que este protocolo não seja atualmente muito usado.

O protocolo Prophet supera o protocolo de encaminhamento epidémico em termos de eficiência, pois já tem em conta o histórico de contactos entre os nós da rede, melhorando, assim, a decisão de encaminhamento em cada nó. O Prophet foi desenhado para ser usado em redes baseadas em comunidades. Este protocolo é também mais eficiente que o Epidemic, pois para além de garantir uma probabilidade de entrega de mensagens parecida, diminui substancialmente o tráfego na rede.

O protocolo SimBet foi dos primeiros protocolos baseados em encaminhamento social e, por isso, só faz sentido compará-lo com o epidémico e o Prophet. Nessa comparação, o SimBet apesar de garantir menos percentagem na entrega de mensagens ao destinatário, é um protocolo mais eficiente visto proporcionar um menor valor de tráfego na rede. Sendo um protocolo antigo, já não é muito utilizado nas RTAs e é substituído por protocolos como o FairRouting, o BubbleRap ou o PeopleRank.

Por sua vez, o protocolo BubbleRap, por usar os dados sociais dos nós da rede na decisão de encaminhamento, mostra-se eficiente na entrega de mensagens, diminuindo o tráfego na rede. Este protocolo é mais eficaz quando a rede tem uma estrutura hierárquica, obtendo, neste caso, uma elevada capacidade de entrega de mensagens e diminuindo o tráfego na rede e nos

nós centrais da mesma. Para além disto, comparando este protocolo com o SimBet, podemos referir que o Bubble tem um melhor desempenho na entrega de mensagens e no custo (tráfego) nas mesmas características, contribuindo, para tal, o fato de ser um protocolo mais recente e, por isso, estar melhorado na performance, em comparação com um mais antigo.

O protocolo FairRouting tem mais em conta a justiça na distribuição de carga, prevenindo o congestionamento de dados nos nós centrais da rede. Comparando este protocolo com o epidémico, podemos verificar um menor valor da percentagem de entrega de mensagens, mas com muito menos carga na rede, o que permite que a eficiência do FairRouting seja muito superior ao epidémico. Em comparação com o SimBet, podemos verificar que o valor da percentagem de mensagens entregues é similar, mas com ligeiramente menos tráfego na rede e substancialmente menos tráfego nos nós centrais, fazendo com que o FairRouting seja superior ao nível da eficiência quando comparado com o SimBet. O FairRouting é mais adequado quando se está preocupado com a justiça na distribuição de carga pelos nós da rede, não sobrecarregando somente os nós centrais.

Por fim, quanto ao protocolo PeopleRank podemos dizer que contem as mesmas características que o FairRouting ou o BubbleRap, sendo que supera o SimBet e o Epidemic quanto à eficiência.

<b>Protocolo</b>	<b>Epidémico</b>	<b>Prophet</b>	<b>SimBet</b>	<b>FairRouting</b>	<b>PeopleRank</b>	<b>BubbleRap</b>
<b>Informação de Contexto</b>	Não tem	Histórico de contactos e transividade	Centralidade, similaridade e força das ligações	Força de interação e assortiveness	Relação social	Divisão entre comunidades e centralidade
<b>Entrega de mensagens</b>	Muito Elevada	Elevada	Razoável	Razoável	S/ Info	Elevada
<b>Atraso</b>	Baixo	Baixo	Relativamente baixo	S/ Info	S/ Info	Baixo
<b>Congestionamento de dados na rede</b>	Muito elevado	Razoável	Razoável	Baixo	S/ Info	Baixo

TABELA II.1 - COMPARAÇÃO PROTOCOLOS

Na Tabela II.1, podemos verificar a comparação dos cinco protocolos estudados em quatro componentes diferentes: informação de contexto, ou seja, as componentes consideradas por cada nó para o encaminhamento de uma mensagem; quantidade relativa no sucesso na entrega de mensagens ao destinatário; o atraso relativo à entrega de uma mensagem; e o nível de congestionamento de dados na rede verificado pela utilização destes protocolos para encaminhamento.

## 2.8 Bundle Protocol

As RTAs são redes com uma arquitetura fim a fim, que permitem comunicações em ambientes altamente stressados, incluindo ambientes com conectividade intermitente, com longos e variados atrasos e com altos valores de erros. O Bundle Protocol (BP) é um protocolo experimental para comunicação nas RTAs feito pelo grupo IRTF's Delay Tolerant Networking Research Group (DTNRG), que divide blocos de dados em pacotes e os transmite usando um mecanismo de *store-and-forward*. As principais características oferecidas pelo BP são: retransmissão baseada em custódia de pacotes (*Custody-based*); capacidade de lidar com a ligação à Internet; capacidade de lidar com ligações previstas, programadas ou oportunistas em adição a ligações contínuas (Internet); uso de identificadores endpoint (endpoint Ids) para definir endereços da Internet constituente.

Este protocolo situa-se na camada aplicacional e funciona em cima de diversos tipos de camadas de transporte e de rede existentes, como podemos observar na Figura II.5.

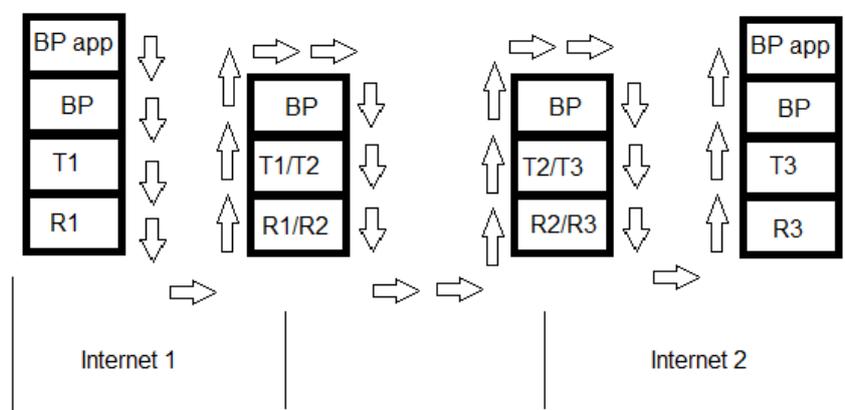


FIGURA II.5 - POSIÇÃO DO BP NA CAMADA OSI

Na Figura II.5, BP app representa uma aplicação Bundle Protocol, Tx representa uma qualquer camada de transporte e Rx uma qualquer camada de rede. A interface responsável por ligar o BP com um outro protocolo de rede é chamada de adaptador da camada de convergência.

A unidade de dados usada no BP é designada de pacote (bundle). Cada pacote é composto por uma sequência de dois ou mais "blocos" de dados do protocolo, que servem vários propósitos. Num pacote de dados existem os dados úteis (o payload) e dados de controlo do protocolo. Os dados podem ser fragmentados em diferentes pacotes, consoante o tamanho do payload a transmitir.

Os nós constituintes de uma rede que usa o BP são denominados de nós Bundle. Cada nó é constituído por três componentes: o agente BP, os adaptadores da camada de convergência e o agente aplicacional. O agente BP é responsável por oferecer os serviços oferecidos pelo BP e executar os procedimentos do BP. Os adaptadores da camada de convergência, como já foi referido, são responsáveis por permitir o uso deste protocolo em diferentes camadas de transporte e de rede e por receber e enviar os pacotes de dados. O agente aplicacional é o componente responsável por utilizar os serviços do BP numa comunicação dentro de um qualquer propósito. O agente aplicacional é dividido em dois elementos: um elemento administrativo responsável por construir e requerer a transmissão de registos administrativos (registos de estados e sinalizações de custódia) e por aceitar e processar sinalizações de custódia que o nó recebe; e por um elemento específico à aplicação que é responsável por cuidar dos dados específicos da aplicação.

Os nós numa rede RTA, segundo o BP, são divididos em endpoints, sendo que um endpoint é composto por zero ou mais nós, todos eles identificados por uma *string* denominada de bundle endpoint ID. A estrutura de cada endpoint ID é revelada a seguir:

< nome do esquema > : < nome específico do esquema ou SSP >

Nenhum dos dois campos deve ultrapassar os 1023 bytes.

O agente BP deve fornecer os seguintes serviços ao agente aplicacional:

- Iniciar um registo de um nó num endpoint;
- Terminar um registo;
- Transmitir um pacote a um determinado endpoint;

- Cancelar uma transmissão;
- Entregar um pacote recebido.

### **2.8.1 Pacotes**

Cada pacote deve ter uma sequência de pelo menos dois blocos. O primeiro bloco em cada pacote tem de ser o bloco primário, sendo este único em todo o pacote. O bloco seguinte ao primário pode ser um bloco referente ao Bundle Security Protocol (BSP) e é opcional. Cada pacote deve ter um bloco de payload. O último bloco de dados de ter uma flag a indicar o fim do pacote.

Os dados no BP são codificados usando o esquema de codificação self-delimiting numeric values (SDNVs). Este esquema é adaptado de Abstract Syntax Notation One Basic Encoding Rules (ASN1). Um SDNV é um valor numérico codificado em N octetos, o último dos quais com o seu bit mais significativo (BMS), definido para zero, o BMS de qualquer outro octeto no SDNV deve ser definido como 1. O valor codificado usando um SDNV é o número binário sem sinal obtido pela concatenação numa única sequência os 7 bits menos significativos de cada octeto do SDNV.

No bloco primário, estão presentes as flags de controlo de processamento do pacote codificadas como um SDNV, sendo que o valor codificado refere-se a uma sequência de bits que invocam uma funcionalidade de controlo do pacote. O mesmo acontece para os outros blocos de dados presentes no pacote, sendo que neste caso as funcionalidades referem-se ao controlo de processamento dos blocos.

No bloco primário, estão também os dados necessários para encaminhar os pacotes para o seu destino. O bloco de payload deve levar o valor de payload dos dados presentes no resto do pacote. O resto dos pacotes leva os dados a serem transmitidos, sendo que cada pacote deve ter os seguintes campos: tipo de bloco, flags de controlo, o tamanho do bloco e os dados propriamente ditos. Cada bloco pode ter, igualmente, uma ou mais referências para Endpoints Ids.

## 2.8.2 Encaminhamento

Todas as transmissões de pacotes são iniciadas em resposta a pedidos dos agentes aplicativos. Podemos dividir a transmissão e encaminhamento de um pacote nas seguintes fases:

1. É pedida a transferência da custódia do pacote a ser enviado, que tem de ser aceite pelo agente BP. Caso o agente BP não permita a transferência de custódia do pacote, este é descartado.
2. A transmissão é iniciada, criando e enviando o pacote inicial com o endpoint ID vazio.
3. O agente BP deve determinar se existe alguma razão para não encaminhar o pacote. Caso não haja nenhuma razão, o agente BP deve escolher o endpoint ID para o destino dos pacotes, sendo que pode ser o destino final do pacote ou um outro nó da rede para futuro encaminhamento. O agente BP deve também escolher o adaptador apropriado da camada de convergência para enviar o pacote.
4. Depois da transferência de custódia ser aceite, o agente BP deve invocar os serviços do adaptador da camada de convergência escolhida e proceder ao envio do pacote.
5. Por fim, o agente BP fica à espera da confirmação da camada de convergência a indicar o sucesso ou insucesso do envio do pacote.
6. Ao receber uma mensagem, o agente BP verifica se o endpoint ID é igual ao seu Endpoint ID e, se for o caso, fica com o pacote, caso contrário, volta ao passo 1.

O BP permite a fragmentação do payload em vários pacotes.

## Sumário

Neste capítulo foram apresentadas várias propostas para a utilização de métricas sociais no encaminhamento em RTAs. No próximo vamos descrever um sistema para explorar este tipo de informação com base em dados sociais retirados de uma rede social online.

# CAPÍTULO III SISTEMA DE RECOLHA E CONSTRUÇÃO DE GRAFOS

O sistema construído para a recolha de dados e construção de grafos sociais e de contactos físicos numa rede RTA, divide-se em três componentes: os nós da rede, o servidor e um sistema de análise dos grafos obtidos. Os nós da rede serão responsáveis por recolher os dados de contactos físicos entre eles e de obter o seu perfil social, enviando a informação obtida para o servidor. O servidor é responsável por receber os dados dos nós e construir um grafo que relacione o histórico de contactos entre os nós e o seu perfil social. O sistema de análise é responsável por calcular as métricas propostas neste documento para relacionar a afinidade social e a relação física entre os nós. O propósito final do sistema é obter dados sobre o histórico de contactos físicos e da relação social entre os nós numa RTA, para posterior análise através das métricas propostas.

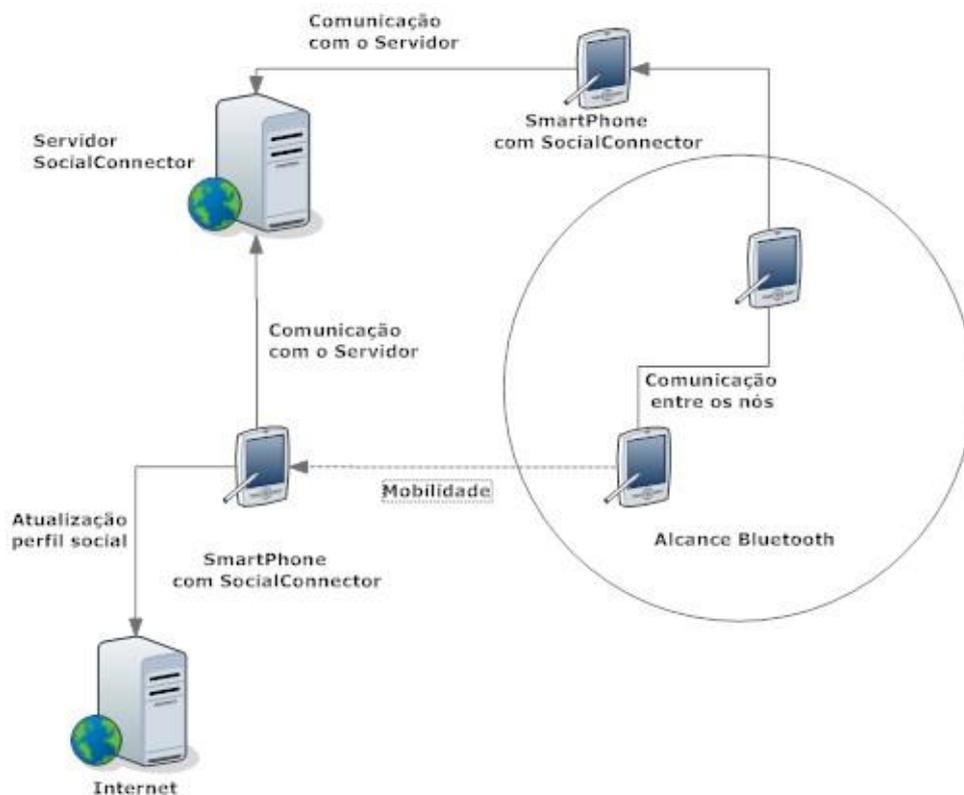


FIGURA III.1 - ESQUEMA GERAL DO SISTEMA DE RECOLHA

Na Figura III.1 podemos observar o esquema da rede RTA formada para recolha dos contactos físicos entre os nós e do seu perfil social. O sistema consiste em três componentes: os nós da rede, responsáveis por recolher os dados dos contactos físicos e sociais; o servidor responsável por receber e armazenar esses dados; e a rede social *online* utilizada para a obtenção do perfil social.

No sistema desenvolvido, foi necessário definir algumas questões como a forma de comunicação entre os nós da rede, o tipo de base de dados utilizada pelo servidor para construir o grafo e a forma de criar o perfil social de cada nó.

### **3.1 Aplicação para a Recolha de Contactos Físicos e Sociais**

Para a realização deste projeto, é necessário obter autorização de um grupo de pessoas para ter acesso ao seu perfil social e ao histórico dos contactos físicos existentes entre eles. Sendo este o objetivo, para persuadir a participação de uma pessoa na experiência decidiu-se criar um incentivo para que as pessoas aceitassem entregar os seus dados - o acesso gratuito a uma aplicação denominada de SocialConnector que visa oferecer benefícios sociais aos seus utilizadores. A aplicação está disponível para telemóveis com o sistema operativo Android e funcionará como um nó da rede RTA.

#### **3.1.1 Escolha da Aplicação**

Depois de um estudo que visou a escolha de uma aplicação que fizesse com que várias pessoas instalassem a aplicação, decidiu-se que a SocialConnector iria consistir numa aplicação que nos mostra os interesses das pessoas que estão ao nosso lado, e que tenham a aplicação instalada, em qualquer local onde nos encontremos. A aplicação funciona à base de avisos, ou seja, quando descobrir alguém na mesma zona que tenha a aplicação instalada, notifica o utilizador da descoberta dessa pessoa, bem como do seu perfil social. A atratividade desta aplicação encontra-se no facto de ajudar as pessoas a socializar, pois permite aos utilizadores poderem descobrir os interesses de uma pessoa desconhecida, facilitando, assim, o início de uma conversa casual e, possivelmente, uma nova amizade. Os interesses dos utilizadores são

obtidos através do Facebook e são atualizados regularmente. Os dados são partilhados entre os utilizadores por Bluetooth.

Estando ligada, a aplicação pode funcionar em três modos diferentes: modo normal, modo egoísta ou modo social. No modo normal, a aplicação tanto está à procura de novos dispositivos para receber os seus perfis, como torna o seu dispositivo visível disponibilizando-se para entregar o seu perfil ao nó encontrado. No modo egoísta, a aplicação procura novos utilizadores unicamente para obtenção de novos perfis, não se tornando visível para os outros nós e não permitindo, assim, enviar o seu perfil a novos utilizadores. No modo social, a aplicação torna-se visível para os outros nós da rede, enviando para estes o seu perfil, mas não procura novos dispositivos para receber perfis.

Sendo esta uma ideia base para a aplicação desejada, posteriormente pensou-se em ideias para o uso desta aplicação noutros contextos. Entre eles, um meio de transporte (avião, camioneta, comboio) ou um evento cultural (teatro, desporto, cinema). A ideia seria obter o conjunto de interesses de cada pessoa e juntar as pessoas com interesses em comum, para que estas possam ter uma conversa interessante. Esta aplicação seria inicialmente comprada por uma empresa interessada (ex: TAP, CP, etc.), que deveria guardar os dados obtidos pela aplicação SocialConnector de cada utilizador e, na altura da marcação dos lugares, juntaria os utilizadores com interesses em comum em lugares próximos. Para a criação desta aplicação, seria necessário acrescentar um plug-in à aplicação SocialConnector que, na altura da marcação do lugar, enviaria para o servidor da empresa, via Internet ou Bluetooth, o seu perfil de interesses, permitindo à empresa comparar os interesses das pessoas e organizar os lugares. Para tal, a empresa deveria ter um servidor que guardasse o perfil de interesses de todos os utilizadores e um programa que implementasse um mecanismo de marcação de lugares que juntasse as pessoas com interesses em comum. Uma outra alternativa seria a própria pessoa a escolher os lugares, sabendo de antemão os interesses das pessoas que já tenham marcado lugar previamente.

Outro contexto onde a aplicação SocialConnector pode ser usada é na descoberta de potenciais pares amorosos. Ou seja, aquando de um encontro entre duas pessoas, a aplicação verificaria a compatibilidade entre elas e, caso tal se confirmasse, indicaria aos dois utilizadores a potencial descoberta de um novo par amoroso. Para implementar esta ideia, era necessário acrescentar à aplicação SocialConnector uma opção onde cada utilizador indicaria se estava à

procura de um novo relacionamento amoroso, e acrescentar, na altura de um encontro oportunista entre dois utilizadores, um algoritmo que verificaria a compatibilidade entre eles. Dois utilizadores seriam compatíveis, caso tivessem a mesma orientação sexual e um conjunto de interesses em comum.

Os nós da RTA criada serão utilizadores que terão instalado nos seus dispositivos móveis (telemóvel) a aplicação SocialConnector e que tem os seguintes requisitos funcionais: definir um protocolo de comunicação para verificar a existência de contactos entre utilizadores; guardar em cache o histórico de contactos até obter acesso ao servidor; enviar o histórico de contactos ao servidor; obter os dados sociais dos utilizadores; enviar os dados sociais dos utilizadores para o servidor; oferecer segurança na troca de dados.

Para responder a estes requisitos a aplicação SocialConnector foi dividida em seis módulos distintos, cada um com a sua responsabilidade como podemos observar na Tabela III.1.

<b>Módulo</b>	<b>Função</b>
Ligação ao Facebook	Responsável por aceder à rede social e retirar os dados desejados, criando um perfil social.
Gestor de perfil	Responsável por gerir o perfil do utilizador, ou seja, definir o que o utilizador pretende partilhar com os outros.
Comunicação entre nós	Responsável por definir e estabelecer o protocolo de comunicação, aquando do encontro entre dois nós.
Armazenamento de dados	Responsável por guardar temporariamente o histórico de contactos do nó e permanentemente o perfil social do utilizador.
Segurança e privacidade	Responsável por garantir a segurança na partilha de informação entre utilizadores e a privacidade dos mesmos.
Ligação ao servidor	Responsável por estabelecer a ligação ao servidor para o envio de dados..

TABELA III.1 - MÓDULOS SOCIALCONNECTOR

Na Figura III.2 podemos observar o esquema da aplicação SocialConnector, onde estão representadas as relações entre os módulos da aplicação.

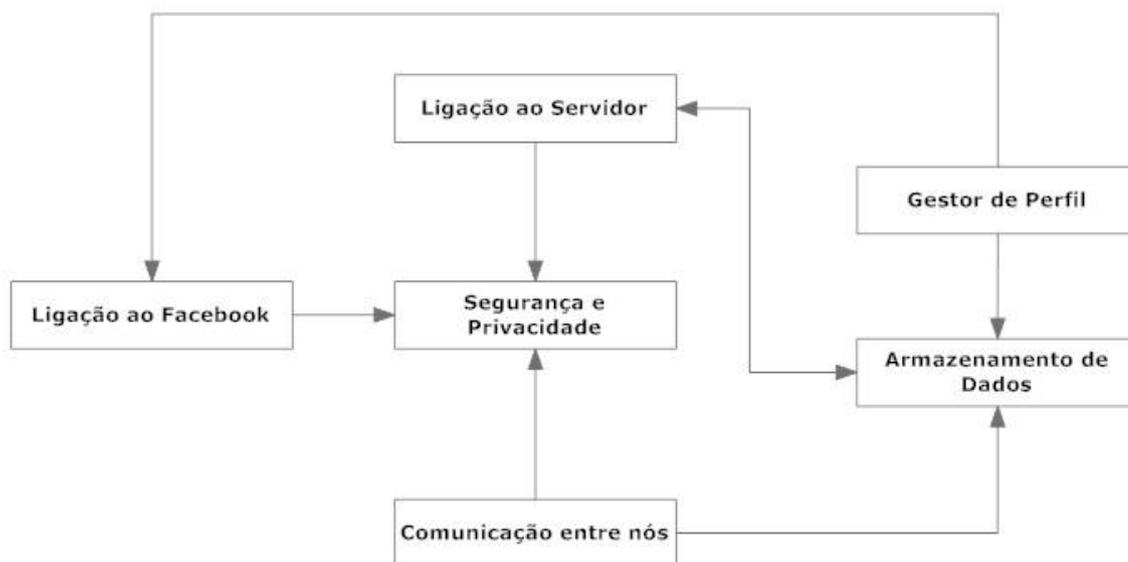


FIGURA III.2 - ESQUEMA SOCIALCONNECTOR

O Gestor de Perfil obtém os dados a partir da ligação ao Facebook, armazenando o perfil social no dispositivo. A comunicação entre os nós é responsável por registar os contactos entre os nós e armazená-los na memória do dispositivo. Aquando de uma ligação ativa à Internet, a aplicação deve comunicar com o servidor de modo a enviar os dados armazenados.

### 3.1.2 Comunicação entre os Nós da Rede

No que concerne a comunicação entre os nós da rede, foram estudadas algumas hipóteses: Bluetooth, infravermelhos, redes ad-hoc Wi-Fi, redes Wi-Fi e redes celulares.

Destas cinco possíveis formas de comunicação, três delas foram automaticamente excluídas por razões distintas. Os infravermelhos foram rejeitados por serem uma tecnologia em decadência, visto já não existirem na maioria dos dispositivos tecnológicos. As redes celulares GSM ou UMTS por terem um custo associado na troca de mensagens não foram consideradas. E por fim, as redes Wi-Fi foram excluídas devido à necessidade de usarem pontos de acesso, o que não permite a possibilidade de comunicação a qualquer momento, um aspeto crucial na rede criada. Como podemos observar em [20], tanto as redes celulares como as redes Wi-Fi

consomem mais energia que o Bluetooth. Com o desenvolvimento da tecnologia é possível que esta diferença diminua e por isso futuramente o Wi-Fi pode ser a melhor opção para a comunicação entre os nós.

<b>Tecnologia</b>	<b>Alcance</b>	<b>Consumo de Energia (mW)</b>	<b>Custo (€)</b>	<b>Desenvolvimento</b>
Bluetooth	Max. 100m em ar livre	Min. 2 Max. 24	Sem custos	Avançado
Wi-Fi	Max. 300m em ar livre	Min. 4.669167 Max. 324.71	Com custos	Avançado
Infravermelhos	Max. 5m em ar livre	S/ info	Sem custos	Descontinuado
Ad-Hoc Wi-Fi	Relativo (depende da potência dos dispositivos envolvidos)	Min. 4.669167 Max. 324.71	Sem custos	Pouco desenvolvido
Redes Celulares	Praticamente ilimitado	Min. 26.7875 Max. 128.6	Com custos	Avançado

TABELA III.2 - COMPARAÇÃO TECNOLOGIAS DE COMUNICAÇÃO

Na Tabela III.2, faz-se uma comparação entre as cinco tecnologias estudadas em quatro campos: o alcance máximo de cada tecnologia de comunicação, o seu consumo de energia mínimo e máximo[20], se tem um custo associado e se a tecnologia está avançada para Android. Da análise da tabela, podemos retirar algumas conclusões: Bluetooth é a tecnologia com o menor valor de gastos energéticos; Wi-Fi tem o maior alcance; Infravermelhos têm pouco alcance e são uma tecnologia descontinuada; as redes celulares e as redes Wi-fi são as únicas tecnologias com custos associados.

Após a análise da Tabela III.2, podemos concluir que duas tecnologias têm vantagem sobre as restantes: Bluetooth e redes ad-hoc Wi-fi. Decidiu-se, portanto, avançar com o Bluetooth, visto que esta tecnologia requer um consumo de energia menor que a utilização do Wi-Fi[18] em dispositivos móveis. Por outro lado, a tecnologia ad-hoc Wi-Fi só está acessível em versões mais recentes do sistema operativo Android o que impede a sua implementação na maioria dos dispositivos.

O Bluetooth é uma tecnologia que permite uma comunicação simples, segura e barata entre vários tipos de dispositivos, tais como computadores, smartphones, telemóveis, ratos, teclados, etc [21]. Utiliza ondas de rádio, ao invés dos cabos que normalmente se utilizam para comunicar, permitindo a troca de informação com uma simples aproximação entre os dispositivos. Utiliza uma combinação de hardware e software que permite a comunicação entre os diversos tipos de dispositivos. Tendo em conta que a comunicação é feita através de radiofrequência, um dispositivo consegue detetar outro, independentemente das suas posições, desde que estejam dentro de um certo limite de distância. O Bluetooth tem um alcance máximo de cem metros (ao ar livre), com uma potência máxima de 100 mW. A velocidade de transmissão de dados pode alcançar, no máximo, 24Mb/s na versão 3.0 do protocolo.

O Bluetooth foi criado para funcionar em todo o mundo, o que implicou a escolha de uma frequência de rádio aberta padrão, ou seja, de uma faixa ISM. A gama escolhida varia dos 2.4 GHz aos 2.5 GHz. A tecnologia Bluetooth usa como esquema de comunicação o FH-CDMA (*frequency hopping – code division multiple access*), que permite proteger a comunicação de interferências, já que divide a frequência em vários canais distintos, fazendo com que a largura de banda seja muito pequena. Dependendo do país, pode utilizar-se até 79 frequências diferentes dentro da faixa ISM no Bluetooth, sendo que cada uma está espaçada por 1 MHz.

Visto que um dispositivo Bluetooth pode transmitir e receber dados (full- duplex), a comunicação é dividida em slots para transmitir e slots para receber, usando, para isso, um esquema FH-TDD (*frequency hopping – time division duplex*), que divide o canal em diferentes intervalos- slots- atribuindo a cada um, uma frequência de salto diferente[22]. O período de cada slot é de 625  $\mu$ s.

Na ligação entre dispositivos, o Bluetooth recorre a dois modos: o SCO (synchronous connection oriented) e o ACL (assynchronous connection-less). No SCO, é estabelecida uma ligação sincronizada entre o dispositivo mestre e o dispositivo escravo, sendo feita uma reserva de *slots* para cada um deles. Por esta razão, este modo é normalmente utilizado quando se pretende um envio contínuo de dados, como por exemplo na comunicação de voz. Este modo não permite a retransmissão de pacotes de dados perdidos e a sua taxa de transmissão é de 432Kb/s, sendo que para voz é de 64Kb/s.

Por seu turno, o modo ACL estabelece uma ligação assíncrona entre o dispositivo mestre e os dispositivos escravos existentes na rede. Permite a retransmissão de pacotes perdidos para

garantir a integridade das informações trocadas. É normalmente utilizado para comunicações que envolvam transferências de ficheiros. A taxa de transmissão de dados neste modo é de 721 Kb/s.

Numa rede Bluetooth, quando dois ou mais dispositivos comunicam entre si, formam uma rede denominada *piconet*. Nessa comunicação, o dispositivo que a iniciou é designado de mestre, enquanto os outros são chamados de escravos. A função do mestre é regular a transmissão dos dados na rede e cuidar do sincronismo entre os dispositivos. Em cada *piconet* podem existir no máximo 8 dispositivos, sendo, ainda assim, possível aumentar este número com a sobreposição de *piconets* – o chamado *scatternet*. Um escravo pode ter esse papel em mais do que uma *piconet*, enquanto que um master o pode fazer em apenas uma *piconet*. É definido um sistema de identificação para que cada dispositivo saiba quais os outros dispositivos que fazem parte da sua *piconet*. Nesse sistema, um dispositivo que queira estabelecer uma ligação numa *piconet* deve emitir um sinal Inquiry. Os dispositivos que recebem esse sinal respondem com um pacote FHS (*frequency hopping synchronization*) informando a sua identificação e os dados de sincronismo da *piconet*. Tendo em base a informação obtida, o dispositivo pode então estabelecer uma ligação, sendo, para isso, necessário enviar um sinal Page. Na Figura III.3 podemos observar o exemplo de uma rede Bluetooth.

Independentemente da tecnologia utilizada como forma de comunicação entre os nós, deve ser possível retirar informação acerca de cada contacto entre dois nós. Os dados relevantes passíveis de serem retirados de um contacto físico entre dois nós são: a identificação dos sujeitos (endereço MAC); a duração do contacto; o instante de início do contacto; a localização geográfica do contacto; e o nível de energia e de memória de cada nó.

Após a análise destes pontos, concluiu-se que alguns campos poderiam ser dispensados, tal como, a localização geográfica do contacto, pois requer o uso de tecnologias que iriam consumir mais recursos aos nós da rede. Para além disso, considera-se o nível de energia e de memória de cada nó dispensável, pois não acrescenta informação quanto à força da ligação entre os nós, sendo que estes valores são importantes unicamente quando tratamos de RTA com nós egoístas.

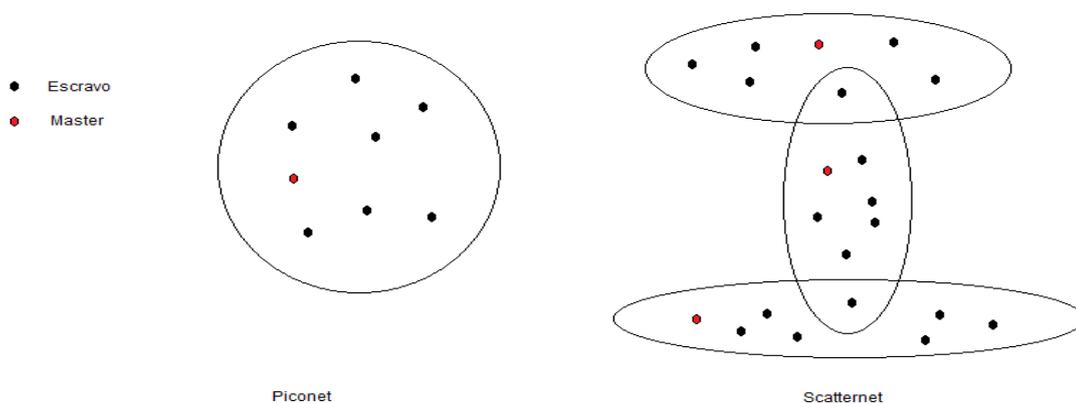


FIGURA III.3 - BLUETOOTH

### 3.1.3 Obtenção do Perfil Social

Uma premissa para a realização deste trabalho foi recolher o perfil social de um grupo de pessoas (os utilizadores da aplicação) para, posteriormente, construir um grafo social. De forma a recolher estes dados, decidiu-se utilizar as redes sociais, visto estarem muito em voga na atualidade e possuírem dados sociais completos de uma grande parte da população. O objetivo é provar a validade da utilização da informação retirada destas redes no encaminhamento em RTAs. Entre as redes sociais, as mais conhecidas e as mais utilizadas são: o Facebook, o Twitter, o MySpace e o LinkedIn.

O Facebook é a rede social online mais utilizada no mundo inteiro, por ser gratuita, de interesse generalizado e simples de usar. Nesta rede social, cada utilizador tem um perfil com a sua informação pessoal, lista de interesses, fotos e uma lista de amigos. Os utilizadores podem trocar mensagens privadas e públicas entre si. Cada pessoa tem um mural onde os seus amigos podem colocar publicações (frases, fotografias, vídeos, etc.) e que pode ser acessado por todos os utilizadores da rede, ou não, dependendo da privacidade definida pelo próprio utilizador. Este tem, também, a possibilidade de criar ou aderir a grupos restritos, identificar amigos em publicações ou fotos, criar publicações no seu próprio mural e “gostar” de publicações. A página principal do Facebook é o *feed* de notícias onde o utilizador pode ver a atividade recente dos

seus amigos e notícias dos seus grupos e das páginas de que gosta. No Facebook, existe, igualmente, a possibilidade de usar aplicações ou jogos desenvolvidos por outros utilizadores.

O Twitter pode descrever-se como um diário online, ou seja, um espaço onde, após a criação de um perfil, se pode publicar mensagens acerca de qualquer tema, permitindo aos seus amigos ver e comentar as publicações. A página principal do Twitter apresenta as mensagens de todas as pessoas que o utilizador segue.

O MySpace é semelhante ao Facebook, pois tem as mesmas características. No entanto, está mais orientado para a cultura, ou seja, para acompanhar as suas bandas musicais preferidas e para mostrar os seus gostos musicais. Tal como no Facebook, cada utilizador tem um perfil com a sua foto, a sua lista de interesses e a sua informação pessoal. Cada utilizador pode pertencer a vários grupos.

O LinkedIn é diferente das redes sociais mencionadas anteriormente, pois visa exclusivamente o relacionamento profissional entre os utilizadores, isto é, permite a procura e oferta de emprego ou contactar um potencial parceiro de negócios. Tal como nas outras redes sociais, é necessário criar uma página de perfil aquando do registo no site com a sua informação pessoal. Posteriormente, o utilizador deve preencher uma página com o seu registo profissional e académico, que servirá como se fosse o seu curriculum. O site tem uma página denominada “contratação e recrutamento”, onde o utilizador pode procurar candidatos por palavra-chave e experiência no ramo pretendido. Para a procura de emprego, o utilizador deve responder a um breve questionário.

Na Tabela III.3, podemos verificar a comparação entre as quatro redes sociais em quatro campos:

- Tipo de relacionamentos: referente ao tipo de relacionamento para que a rede social foi criada.
- Grupos: se a rede social possui a possibilidade da criação de grupos.
- Chat: se a rede social oferece a possibilidade de falar online com os seus amigos. De referir que, neste caso, algumas das redes sociais permitem chat mas precisam da instalação de uma aplicação e, por essa razão, na tabela está indicado como se não tivessem.
- Aplicações: se é possível usar ou desenvolver aplicações dentro da rede social.

- Utilizadores: refere-se ao número de utilizadores que visitam a rede social mensalmente, ou seja, utilizadores ativos da rede social segundo [23].

<b>Rede Social</b>	<b>Tipo de Relacionamentos</b>	<b>Grupos</b>	<b>Chat</b>	<b>Aplicações</b>	<b>Utilizadores (milhões)</b>
Facebook	Pessoais e Profissionais	Sim	Sim	Sim	750
Twitter	Pessoais	Não	Não	Não	250
MySpace	Pessoais	Sim	Não	Sim	70.5
Linkedin	Profissionais	Não	Não	Não	110

TABELA III.3 - COMPARAÇÃO REDES SOCIAIS

Analisando a Tabela III.3, podemos verificar que a rede social mais completa e consequentemente a mais usada em todo o mundo é o Facebook.

Como foi referido anteriormente, o objetivo da criação desta aplicação serve como incentivo para recolher dados sobre contactos existentes e sobre relações sociais entre um grupo restrito de pessoas previamente definidas. O objetivo final será criar dois grafos: um grafo social e um grafo de contactos físicos, e verificar se existe uma verdadeira relação entre eles. Pretende-se saber, então, se pessoas que têm uma forte ligação social, ou seja, que partilham os mesmos interesses e têm amigos em comum, têm mais hipóteses de se encontrar do que pessoas que tenham uma menor ligação social.

Para atingir este objetivo, foi necessário definir um conjunto de dados passíveis de serem retirados do Facebook, tais como, identificação do utilizador (nome e id); lista de amigos; lista de interesses; lista de familiares; localização; número de identificações em fotos/posts de amigos; número de feeds/mensagens trocadas com amigos.

Tendo em conta que o Facebook SDK tem algumas limitações na recolha de dados, e visto que se pretende preservar a máxima privacidade possível do utilizador, decidiu-se prescindir de obter os dois últimos campos, ou seja, o número de identificações em fotos/posts de amigos e o número de feeds/mensagens trocadas com amigos.

### 3.2 Servidor para Armazenamento dos Dados

O servidor é uma aplicação desenvolvida em Java que deve estar registada na Internet para poder comunicar com os clientes. O servidor tem os seguintes requisitos funcionais: receber o perfil social e o histórico de contactos dos nós; permitir adicionar e remover novos nós; construir os grafos sociais e de contactos físicos; permitir o acesso aos clientes através da Internet; garantir a privacidade dos utilizadores na apresentação de resultados; garantir segurança na troca de informação com os utilizadores.

Para responder a estes requisitos, o servidor está dividido em cinco módulos que estão demonstrados na Tabela III.4, onde indicamos também a função de cada um.

<b>Módulo</b>	<b>Função</b>
Registo na Internet	Responsável por colocar o servidor disponível na Internet, de modo a que esteja acessível aos nós da rede.
Comunicação com os nós	Responsável por definir um protocolo de comunicação com os nós da rede, de modo a receber periodicamente o histórico de contactos e o perfil social de cada nó.
Processamento dos dados obtidos	Responsável pelo processamento das mensagens recebidas dos nós da rede, de modo a armazenar a informação num grafo NoSQL.
Criação dos grafos	Responsável por guardar permanentemente os dados obtidos num grafo.
Segurança e privacidade	Responsável por garantir a segurança na partilha de informação com os nós da rede e por definir um mecanismo de segurança que permita garantir a privacidade dos utilizadores na apresentação dos dados.

TABELA III.4 - MÓDULOS SERVIDOR

Na Figura III.4 podemos observar o esquema do servidor com a relação existente entre os seus módulos.

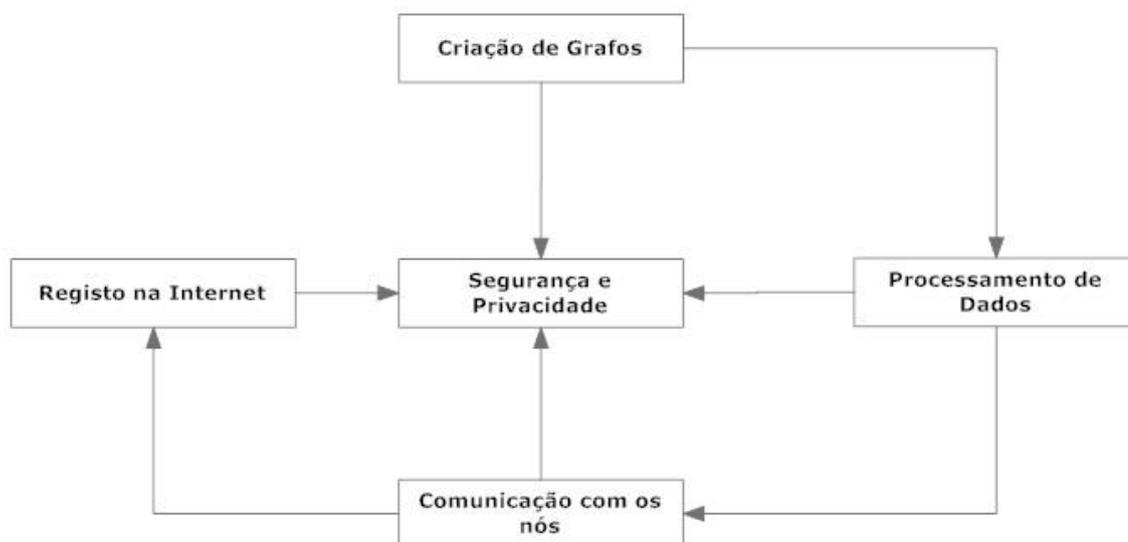


FIGURA III.4 - ESQUEMA SERVIDOR

Depois de comunicar com os nós, através de uma ligação segura na Internet, o servidor obtém os dados dos seus encontros, fazendo o seu processamento e armazenamento em grafos para posterior análise.

### 3.2.1 Registo do Histórico de Interações

Para o desenvolvimento deste projeto, foi necessária a construção de uma base de dados pelo servidor de forma a guardar os dados de contactos físicos entre os nós da rede e o perfil social desses mesmos nós. Para esse efeito, foi necessário escolher um tipo de base de dados, sendo que essa escolha poderia recair sobre uma base de dados relacional (SQL) ou uma base de dados não relacional (NoSQL) [24] [25].

Visto que, para este projeto, o que pretendemos guardar são dados obtidos de uma rede, o tipo de base de dados mais indicado seria uma base de dados não relacional, ou seja, a base de dados NoSQL.

O termo NoSQL (Not only SQL) refere-se a uma classe de base de dados não relacional, acabando, desta forma, com o uso exclusivo de bases de dados relacionais. A principal razão do aparecimento deste tipo de base de dados deve-se à necessidade crescente de obter maior

escalabilidade no armazenamento de dados. As bases de dados relacionais também promovem escalabilidade, no entanto, quanto maior for o tamanho dos dados, mais custosa se torna a escalabilidade, seja pelo custo de novas máquinas ou pela manutenção de especialistas no local onde se encontram as bases de dados. Por esta razão, foram criadas as bases de dados não relacionais NoSQL, que permitem uma escalabilidade mais barata e menos trabalhosa, visto não necessitarem de máquinas muito poderosas e a sua manutenção ser simples, permitindo assim reduzir o número de profissionais necessários.

Estas bases de dados estão a ficar cada vez mais populares entre as grandes empresas, pois têm características que permitem trabalhar com dados semiestruturados vindos de diversas origens (arquivos de logs, websites, multimédia, etc.). Podemos enumerar algumas dessas características: processamento paralelo de informações dividindo tarefas por diferentes processadores; distribuição em escala global utilizando para isso diversos tipos de bases de dados não relacionais.

Existem alguns tipos de bases de dados não relacionais: bases de dados chave/valor, bases de dados orientados a documentos e bases de dados de grafos. Enquanto o primeiro e segundo tipo não são indicados para usar com grafos, o último tipo de bases de dados é bastante compatível com o nosso caso já que estamos a tratar de redes que, por sua vez, podem ser representadas por grafos [26]. Este tipo de base de dados não relacionais, ao contrário de outros tipos de bases de dados, está diretamente relacionado com um modelo de dados pré estabelecido, o modelo de grafos. A ideia é representar os dados como grafos dirigidos. Este tipo de modelo funciona melhor quando se pretende dar mais relevância à conectividade ou à topologia dos dados que aos dados propriamente ditos. É constituído por três elementos básicos: os nós (vértices do grafo), as relações (as arestas) e as propriedades dos nós e das relações. Algumas das bases de dados que utilizam o modelo de grafos são: Neo4j, Infinite Graph, InforGrid, HyperGraphDB, etc. No âmbito deste projeto, vai ser utilizada a base de dados Neo4j que está desenhada para ser usada na linguagem de programação Java[27]. Esta base de dados foi criada pela Neo Technology e o seu uso é bastante fácil e intuitivo, proporcionando uma forma eficaz de representar bases de dados por grafos e dando a possibilidade de atribuir, de uma forma simples, atributos tanto aos nós do grafo, como às ligações entre eles. Esta característica é bastante interessante para o projeto, pois o que se pretende fazer é criar um grafo social em que os nós da rede e as suas ligações representam pessoas e as relações

sociais entre elas, bem como os dados dos contactos físicos entre si (caso tenham existido). Outra das características desta base de dados consiste em permitir descobrir o melhor caminho de um nó para outro utilizando algoritmos diferentes (Dijkstra, A\*).

### 3.3 Análise dos Dados

O objetivo final desta tese científica consiste em obter a relação entre o histórico de contactos físicos e a relação social dos nós numa rede RTA, através do uso de coeficientes, que poderão ser utilizados como uma ajuda na decisão de encaminhamento numa RTA.

Com esse propósito, foram definidas duas métricas: o coeficiente social entre o nó A e o nó B  $C_{SAB} \in [0,1]$  ; e o coeficiente dos contactos físicos entre o nó A e o nó B  $C_{FAB} \in [0,1]$ . O coeficiente social é calculado da seguinte forma:

$$C_{SAB} = \partial \times \text{Amigo}_{AB} + \sigma \times C_{AAB} + \beta \times C_{IAB} \quad (24)$$

Nesta métrica, estão incluídos o valor  $\text{Amigo}_{AB}$ , o coeficiente de amizade  $C_{AAB} \in [0,1]$  e o coeficiente de interesses  $C_{IAB} \in [0,1]$ . O valor  $\text{Amigo}_{AB}$  indica se A pertence à lista de amigos de B, sendo que toma o valor 1 caso pertença e o valor 0 caso contrário. O coeficiente de amizade indica a razão entre o número de amigos em comum entre A e B, sobre o número total de amigos de A, uma vez que estamos a calcular o coeficiente para a ligação de A para B. O segundo, refere-se à razão entre o número de interesses em comum entre A e B, sobre o número total de interesses de A.

$\partial + \beta + \sigma = 1$  são valores que se referem à importância que se pretende dar a cada coeficiente. Nas Equações 25, 26 e 27 é mostrado como é efetuado o cálculo destes coeficientes.

$$C_{AAB} = \frac{(N^{\circ} \text{ Amigos em comum})_{AB}}{(N^{\circ} \text{ Total amigos})_A} \quad (25)$$

$$C_{I_{AB}} = \frac{(N^{\circ} \text{ Interesses em comum})_{AB}}{(N^{\circ} \text{ Total interesses})_A} \quad (26)$$

$$\text{Amigo}_{AB} = \begin{cases} 0, & \text{se } B \notin \text{ListaAmigos}_A \\ 1, & \text{se } B \in \text{ListaAmigos}_A \end{cases} \quad (27)$$

O coeficiente de contactos físicos é calculado da seguinte forma:

$$C_{F_{AB}} = [\delta \times C_{C_{AB}} + \alpha \times C_{D_{AB}}] \times \rho^{(\text{tempo médio espera})_{AB}} \quad (28)$$

Em relação ao coeficiente dos contactos físicos, este consiste, igualmente, em dois coeficientes: o coeficiente do número de contactos físicos  $C_{C_{AB}} \in [0,1]$ ; e o coeficiente da duração desses mesmos contactos físicos  $C_{D_{AB}} \in [0,1]$ . O primeiro, alude à razão entre o número de contactos entre os nós A e B na rede, sobre o número total de contactos do nó A com todos os nós da rede. O segundo, aponta a razão entre a duração total dos contactos entre A e B, sobre a duração total dos contactos de A com todos os nós da rede. O coeficiente físico tem também um campo de envelhecimento,  $\rho^{(\text{tempo médio espera})_{AB}}$ , que faz com que o valor diminua consoante o  $(\text{tempo médio espera})_{AB}$  aumente.  $\delta + \alpha = 1$  são valores que se referem à importância que se pretende dar a cada coeficiente.  $\rho \in [0,1]$  é a constante de envelhecimento; e  $(\text{tempo médio espera})_{AB}$  é a média de tempo decorrido entre os contactos dos nós A e B. Nas Equações 29 e 30 é mostrado como se faz o cálculo dos coeficientes descritos.

$$C_{C_{AB}} = \frac{(N^{\circ} \text{ Contactos})_{AB}}{(N^{\circ} \text{ Total contactos})_A} \quad (29)$$

$$C_{D_{AB}} = \frac{(\text{Duração contactos})_{AB}}{(\text{Duração contactos})_A} \quad (30)$$

Por fim, para relacionar os dois coeficientes fazemos a razão entre eles e retiramos as devidas ilações.

$$Amizade_{AB} = \frac{C_{F_{AB}}}{C_{S_{AB}}} \quad (31)$$

$$\left\{ \begin{array}{l} Amizade_{AB} \approx 1 \rightarrow \text{relação social} \approx \text{relação física} \\ Amizade_{AB} \ll 1 \rightarrow +\text{relação social} \\ Amizade_{AB} \gg 1 \rightarrow +\text{relação física} \end{array} \right. \quad (32)$$

A métrica  $Amizade_{AB}$  apresentada na Equação 31, tem o intuito de demonstrar se existe uma relação entre o histórico de contactos físicos e a afinidade social entre os nós numa RTA. No CAPÍTULO V é analisada esta métrica utilizando dados de um *dataset* previamente construído.

Uma aplicação para o uso dos coeficientes apresentados neste trabalho consiste na utilização dos mesmos como apoio à decisão de encaminhamento para um protocolo RTA. A decisão de encaminhamento seria então realizada da seguinte forma: cada nó da rede guardava o seu perfil social e histórico de contactos físicos com os outros nós. Aquando do encontro de um novo nó, essa informação seria trocada entre os nós de modo a cada um deles poder calcular o coeficiente físico e social em relação ao outro. Como temos dois coeficientes, o físico e o social, deve-se realizar a combinação dos mesmos sob a forma da Equação 33 de modo a obter um único valor para a relação entre os dois nós que relacione a sua afinidade social com o histórico de contactos físicos:

$$C_{C_{AB}} = \omega \times C_{F_{AB}} + \varphi \times C_{S_{AB}} \quad (33)$$

Onde  $\omega + \varphi = 1$  são valores que se referem à importância que se pretende atribuir ao campo social e físico.

Depois de calculados os coeficientes, estes deviam ser armazenados nos nós sob a forma de um vector, de modo a um nó poder manter na sua memória informação acerca dos coeficientes sobre todos os outros nós da rede. Essa matriz também seria trocada pelos nós aquando de um novo encontro entre eles. Posteriormente ao cálculo e armazenamento dos

coeficientes, cada nó devia listar a sua lista de mensagens pendentes, e, se o nó encontrado tiver um valor da soma dos coeficientes físico e social superior ao seu para o destino da mensagem, esta seria entregue ao nó encontrado.

A cada novo encontro entre dois nós que já se tenham encontrado anteriormente o valor dos coeficientes é atualizado.

### **Sumário**

Neste capítulo foi descrito o sistema e propuseram-se métricas novas para utilização na decisão de encaminhamento num protocolo numa RTA. No capítulo seguinte será descrita a forma como foi feita a implementação do sistema.

## CAPÍTULO IV IMPLEMENTAÇÃO

O sistema implementado baseia-se num esquema Cliente-Servidor, onde o cliente é responsável pela recolha de dados e o servidor pela receção e armazenamento dos mesmos. O cliente do nosso sistema é representado por uma aplicação para dispositivos Android, a aplicação SocialConnector e o servidor é uma aplicação Java para Windows. Neste capítulo será explicado como se realizou a implementação destas duas aplicações.

### 4.1 Aplicação SocialConnector

Como foi referido anteriormente, cada nó da rede terá a aplicação SocialConnector instalada de modo a que se possa obter o histórico dos contactos físicos e dos dados sociais de cada nó para, posteriormente, enviar para o servidor. A aplicação tem um menu inicial, onde existe a possibilidade de escolher uma opção entre quatro possíveis. Na Tabela IV.1 podemos observar essas opções.

<b>Opção</b>	<b>Função</b>
Meu Perfil	Ver e alterar dados perfil social
Lista Perfis Recentes	Ver perfis encontrados recentemente
Alterar Modo Aplicação	Alterar modo de funcionamento da aplicação
Definições	Alterar as definições

TABELA IV.1 - MENU INICIAL

Escolhendo a opção Meu Perfil, a aplicação mostra os dados do perfil social do utilizador, permitindo-lhe atualizar os dados do seu perfil através do Facebook. A Lista de Perfis Recentes apresenta-nos uma lista de pessoas que a aplicação encontrou recentemente, onde podemos observar ao perfil social de cada uma delas. No Alterar Modo de Aplicação, é possível definir o modo em que se pretende que a aplicação funcione. Nas definições, o utilizador pode alterar as definições da aplicação.

De seguida, será explicado o funcionamento dos seis módulos que compõe esta aplicação.

#### 4.1.1 Gestor de Perfil e Ligação ao Facebook

A aplicação SocialConnector deverá ter acesso aos dados do Facebook de cada utilizador e, para tal, terá que pedir a permissão ao utilizador para ter esse acesso aquando da instalação da mesma, como podemos observar na Figura IV.1. O utilizador deverá estar ligado à Internet no momento da obtenção de dados do Facebook, ou seja, na primeira utilização da aplicação, de forma a poder criar o seu perfil.



FIGURA IV.1 - DIAGRAMA OBTENÇÃO PERFIL SOCIAL

As permissões de acesso ao Facebook que a aplicação requiere são as seguintes: nome do utilizador, data de nascimento, localização, lista de amigos e lista de interesses. Para obter esses dados, recorreu-se à ferramenta Facebook SDK integrada no eclipse. Também é possível, sempre que o utilizador pretender, atualizar os dados de perfil do utilizador através do Facebook. Assim, se entretanto o utilizador alterar algum campo no Facebook, tal como adicionar ou remover um amigo ou um interesse, pode aceder à página de gestão de perfil na aplicação, e desde que tenha uma ligação ativa na Internet, atualizar os seus dados através do Facebook.

Ao retirar os dados do Facebook, o utilizador cria um perfil onde irá ter a sua informação para partilhar com os outros utilizadores. Devido a questões de privacidade, na aplicação é permitido ao utilizador escolher que dados do seu perfil pretende trocar com os outros dispositivos, sendo possível esconder, se pretendido, alguns dos seus interesses ou dos seus amigos. Esta opção pode ser feita a qualquer momento, sendo realizada pressionando o interesse ou amigo que pretende esconder/mostrar e escolher a opção *esconder/mostrar* consoante o seu estado anterior.

Cada utilizador vai ter, portanto, um perfil para trocar com outros utilizadores. Na Figura IV.2 apresenta-se o exemplo de um perfil na aplicação.

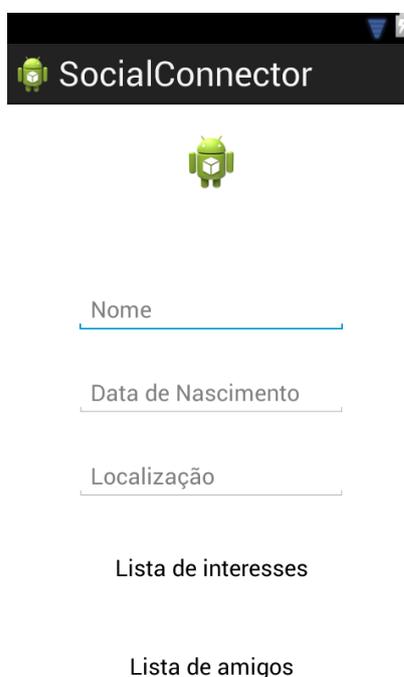


FIGURA IV.2 - PERFIL DO UTILIZADOR

O nome, a idade e a localização são campos básicos que não requerem nenhum tratamento especial, sendo apresentados diretamente na página de perfil. Em relação à lista de interesses e à lista de amigos, é necessário que sejam retiradas e guardadas em objetos especiais como *HashMaps*. Aqui, para além de se guardarem as designações dos interesses e dos amigos, guardam-se igualmente os seus identificadores retirados do Facebook para, posteriormente, poder compará-los com os interesses e amigos dos outros utilizadores visto

poder haver a mesma designações para amigos ou interesses diferentes, já os identificadores são únicos. Tanto a lista de interesses como a lista de amigos são botões que, quando carregados, apresentam por ordem alfabética a lista de interesses e amigos respetivamente.

A foto de perfil é retirada da foto de perfil do Facebook, mas pode ser alterada por outra que o utilizador tenha no dispositivo que está a utilizar.

#### 4.1.2 Comunicação entre Nós

Como já foi referido a aplicação, pode funcionar em quatro modos distintos, tal como pode ser observados na Tabela IV.2.

<b>Modo</b>	<b>Procura dispositivos</b>	<b>Visibilidade</b>
Desligado	Não	Não visível
Normal	Sim	Visível
Social	Não	Visível
Egoísta	Sim	Não Visível

TABELA IV.2 - MODOS APLICAÇÃO

No modo desligado, a aplicação tem o Bluetooth desligado e, por isso, nem procura novos dispositivos para obter o seu perfil, nem está disponível para partilhar o seu perfil com outros utilizadores. No modo normal, a aplicação procura novos dispositivos para obter os seus perfis sociais e torna-se visível para os outros utilizadores, de modo a poder fornecer-lhes o seu perfil social. No modo social, não procura novos dispositivos, disponibilizando-se unicamente a enviar o seu perfil a novos utilizadores. No modo egoísta, procura novos utilizadores para obter o seu perfil social, mas não se torna visível, o que não permite que outros utilizadores obtenham o seu perfil. Com o propósito de funcionar nestes três modos, a aplicação deve permitir procurar novos dispositivos, ao mesmo tempo que permite ao nó ficar visível para outros. A comunicação funciona da seguinte forma: se um nó A encontrar um nó B, ou vice-versa, depois de verificar se

o nó B tem a aplicação instalada e a funcionar no modo normal ou social, o nó A pede uma parte do perfil do nó B, com o identificador, o seu nome, idade e localização.

Depois de receber esses primeiros dados, o nó A deve decidir se pretende pedir o resto do perfil social, e caso o pretenda deve pedi-lo ao nó B. O nó B deve dar permissão para enviar o resto do perfil social. Caso o utilizador B permita o envio do perfil, é enviado para o nó A a totalidade do seu perfil: a sua foto, a sua lista de interesses e a lista de amigos.

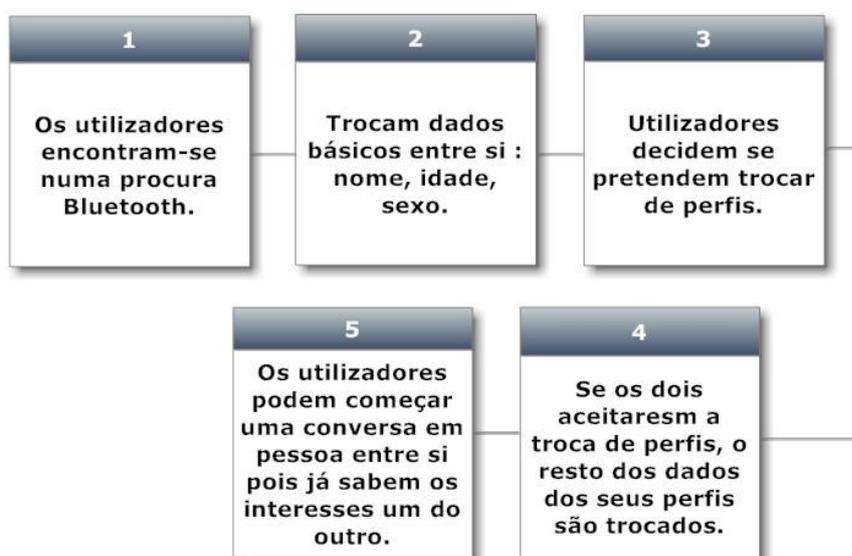


FIGURA IV.3 - ENCONTRO ENTRE UTILIZADORES

Na Figura IV.3 observa-se os passos realizados aquando de um encontro entre dois utilizadores.

Como a comunicação é feita através do Bluetooth, cada nó da rede deve funcionar tanto como servidor, como cliente. Como servidor, torna o dispositivo visível para os outros nós e fica à espera de ligações de um cliente; como cliente, procura novos dispositivos e quando encontra um, inicia uma conversa. Na Figura IV.4 apresenta-se o protocolo de comunicação detalhado entre dois nós que se encontram na rede, sendo que um deles, o nó B, está a funcionar como servidor Bluetooth (mestre), disponível para ceder o seu perfil social; e outro como cliente (escravo), que está à procura de novos utilizadores para obter o seu perfil social. O tempo entre procura de dispositivos é definido pelo utilizador nas definições, podendo tomar três valores

distintos: longa (1 minuto), média (30 segundos) ou curta (10 segundos). Cada procura de dispositivos tem uma duração média de 12 segundos.

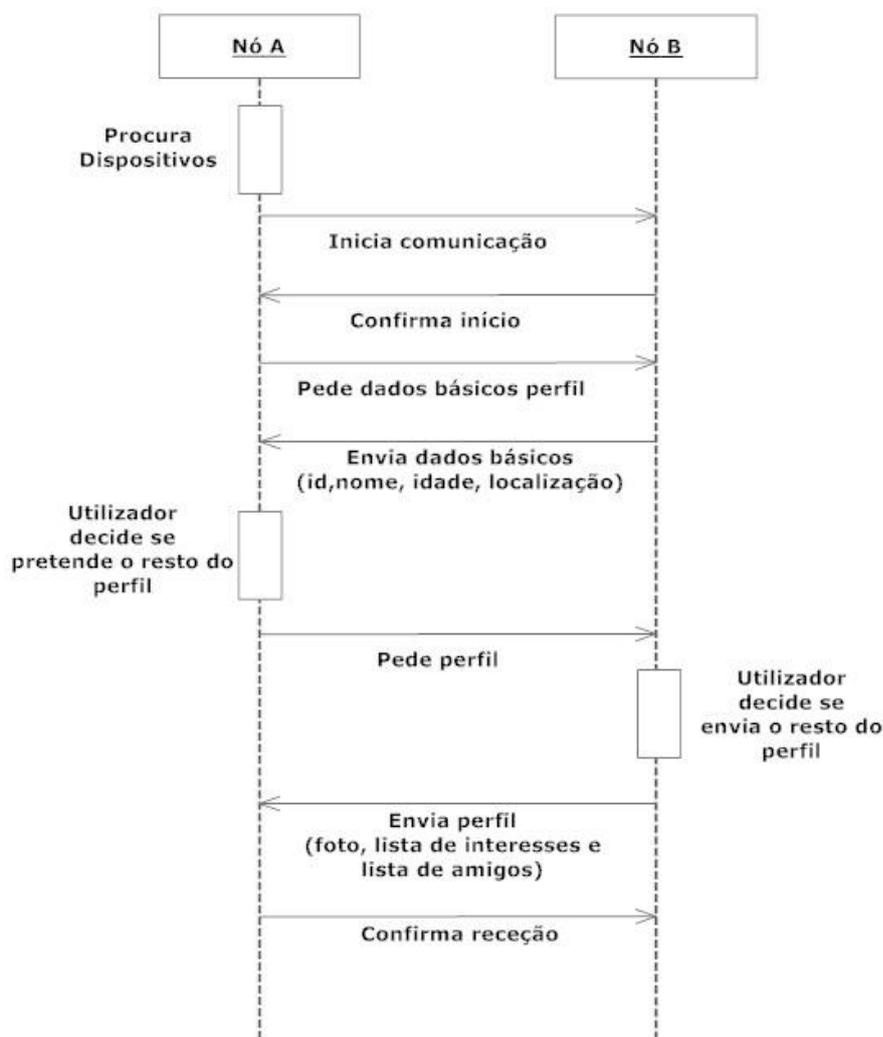


FIGURA IV.4 - PROTOCOLO COMUNICAÇÃO ENTRE OS NÓS

A comunicação entre os nós é feita usando uma ligação RFCOMM segura, com registo do serviço Bluetooth com o UUID "00001101-0000-1000-8000-00805F9B34FB". Este tipo de ligação é fornecido pelo Android através de APIs[28].

No fim desta troca de mensagens, é acrescentado o perfil encontrado na lista de perfis recentes.

Aquando de um novo encontro, e depois da partilha de perfis entre si, os nós devem comunicar e, caso pretendam, devem enviar a lista de perfis encontrados recentemente para, desta forma, contrariar o limite de alcance do Bluetooth, utilizando a transitividade na partilha de

perfis. Para salvaguardar a privacidade do utilizador no fim do protocolo da Figura IV.4, é realizada uma troca de mensagens onde se pergunta ao nó que cede o perfil se aceita que o seu perfil seja passado a outros utilizadores, caso este aceite, o seu perfil é enviado a cada novo encontro, caso contrário o perfil é guardado na lista de perfis recentes, mas não é enviado para os outros nós.

O tamanho da lista de encontros recentes é definido pelo utilizador nas definições, onde este pode escolher entre três opções: guardar lista comprida (perfis encontrados nos últimos 30 minutos), média (perfis encontrados nos últimos 15 minutos) ou curta (perfis encontrados nos últimos 5min). A lista de perfis recentes é listada na opção do menu inicial Mostrar Perfis Recentes, onde, em cada perfil social, podemos obter as seguintes informações: nome, idade, localização, lista de interesses, lista de amigos, lista de interesses em comum e lista de amigos em comum. No diagrama da Figura IV.5 podemos verificar como funciona a decisão de envio destes perfis entre os nós. Esta funcionalidade permite aos nós da rede conhecer o interesse de novas pessoas, sem entrar em contacto próximo com elas, utilizando, para isso, as características da nossa RTA.

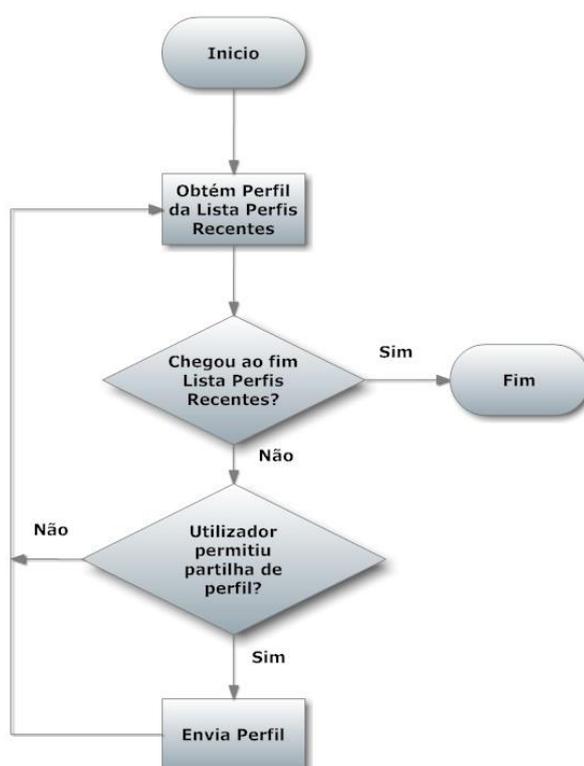


FIGURA IV.5 - TROCA LISTA PERFIS

A comunicação para o envio de cada perfil é feita usando o mesmo protocolo da Figura IV.4.

### **4.1.3 Recolha e Armazenamento de Dados**

Todos os dados importantes da aplicação são armazenados no cartão de memória SD, se disponível, ou então na memória do telefone. Assim, mesmo que o dispositivo se desligue ou fique sem bateria, os dados são preservados. Para o efeito, é criada uma pasta SocialConnector onde se guardará toda a informação.

Aquando da primeira ligação ao Facebook, ou numa atualização do perfil social, os dados são guardados consoante a informação, ou seja, o nome, idade, localização, lista de interesses e lista de amigos serão armazenadas em *Strings* das *Shared Preferences* do dispositivo *Android*, sendo que a foto gravada na pasta SocialConnector, será guardada num ficheiro PNG com nome igual ao identificador do utilizador.

Relativamente aos contactos com os outros nós da RTA, deve ser guardada uma lista de encontros com a seguinte informação: identificador da ligação, início do encontro, fim do encontro e identificador do dispositivo encontrado. O início do encontro é obtido aquando do início da comunicação com o dispositivo encontrado. O fim do encontro é o tempo da última procura de dispositivos em que o nó foi descoberto, antes da inexistência de resposta do nó entre dois períodos consecutivos de procura. Ou seja, considera-se que o contacto entre os dois nós termina quando um dos nós deixa de aparecer em duas procuras de dispositivos consecutivas. Todos os tempos são obtidos através do relógio do dispositivo e são tempos relativos em milissegundos desde 1 de Janeiro de 1970.

Com o intuito de no servidor relacionar o mesmo contacto entre nós, cada nó deve manter um identificador para cada novo encontro com outro nó. O identificador de ligação é obtido através do protocolo de comunicação da Figura IV.4. Com esse propósito, no início da comunicação da Figura IV.4 o nó A, na mensagem de início de comunicação, envia uma proposta para um identificador que seja único na sua lista de encontros, o nó B verifica se esse identificador é também único na sua lista de encontros, e caso seja, envia a mensagem de confirmação de início, em caso contrário envia uma mensagem de erro para o nó A e este gera novos identificadores até que o identificador gerado seja único nos dois nós.

A informação da lista de encontros é mantida na memória do dispositivo até haver uma ligação com o servidor. Após os encontros serem enviados para o servidor, a sua informação é retirada da memória do dispositivo.

#### 4.1.4 Segurança e Privacidade

Um dos problemas desta aplicação reside na privacidade dos dados trocados entre os utilizadores. Nenhum utilizador deseja que os seus dados e interesses sejam partilhados livremente, sem ter a segurança de que os seus dados são salvaguardados, ou seja, não sejam usados para um objetivo que não seja permitido pelo utilizador. Por isso, no momento da instalação da aplicação, os utilizadores são informados acerca das questões de privacidade que podem advir do uso desta aplicação.

Para salvaguardar a privacidade do utilizador, como foi descrito anteriormente, este escolhe o que deseja trocar com os outros utilizadores, permitindo assim mostrar só os dados que pretender.

Para além disso, antes de serem enviados para o servidor, a aplicação usa um mecanismo de *hashing* SHA-1[29] em todos os identificadores retirados do Facebook, o que leva a que a informação pessoal de cada utilizador seja salvaguardada, uma vez que nenhuma informação pessoal será efetivamente enviada para o servidor, só os seus identificadores calculados através do mecanismo de hash. Um *hash* SHA-1 é criado, tendo como entrada uma *String* com tamanho máximo de 256 bits, neste caso os identificadores Facebook, e como saída temos uma impressão digital de 160 bits. Codificando a mesma *String* usando o algoritmo SHA-1, obtém-se sempre a mesma saída de *hash*.



FIGURA IV.6 - HASHING SHA-1

A Figura IV.6 demonstra o funcionamento do mecanismo de *hashing* SHA-1. Existe uma probabilidade muito pequena de o identificador hash SHA-1 não ser único e, por isso, é possível substituir os identificadores obtidos do Facebook pelos identificadores SHA-1 calculados.

Para garantir a segurança na comunicação com o servidor, são utilizados sockets SSL com uso de certificados.

#### **4.1.5 Comunicação com o Servidor**

Cada vez que é acessada no dispositivo Android, a aplicação SocialConnector verifica se o dispositivo tem uma ligação ativa à Internet. Caso haja ligação, e houver alguma informação para enviar, há uma tentativa de ligação ao servidor, disponível através de um IP pré-definido. Ao ligar-se ao servidor, a aplicação pode enviar três tipos de mensagens: *novo\_encontro*, *perfil\_social* ou *atualização\_perfil\_social*.

A mensagem *perfil\_social* é usada na primeira ligação ao servidor e serve para a aplicação enviar os dados do perfil social criado para o servidor. Esta mensagem tem os seguintes campos: identificador do utilizador, a idade, o identificador da localização, a lista de identificadores de interesses que o utilizador permitiu partilhar e a lista de identificadores de amigos que o utilizador permitiu partilhar. Todos os identificadores enviados foram obtidos através da função de hash.

A mensagem *atualização\_perfil\_social* é usada quando o utilizador faz uma atualização do seu perfil social desde a última comunicação com o servidor. A mensagem tem os seguintes campos: a nova lista de identificadores de interesses que o utilizador permitiu partilhar e/ou a nova lista de identificadores de amigos que o utilizador permitiu partilhar.

A mensagem *novo\_encontro* é enviada percorrendo a lista de encontros da aplicação e enviando cada um dos encontros separadamente. Uma mensagem deste tipo tem os seguintes campos: id do utilizador, id do utilizador encontrado, id do encontro, tempo de início do encontro e tempo de fim do encontro. Após receber a confirmação do servidor da receção da mensagem, o encontro é eliminado da memória local do dispositivo.

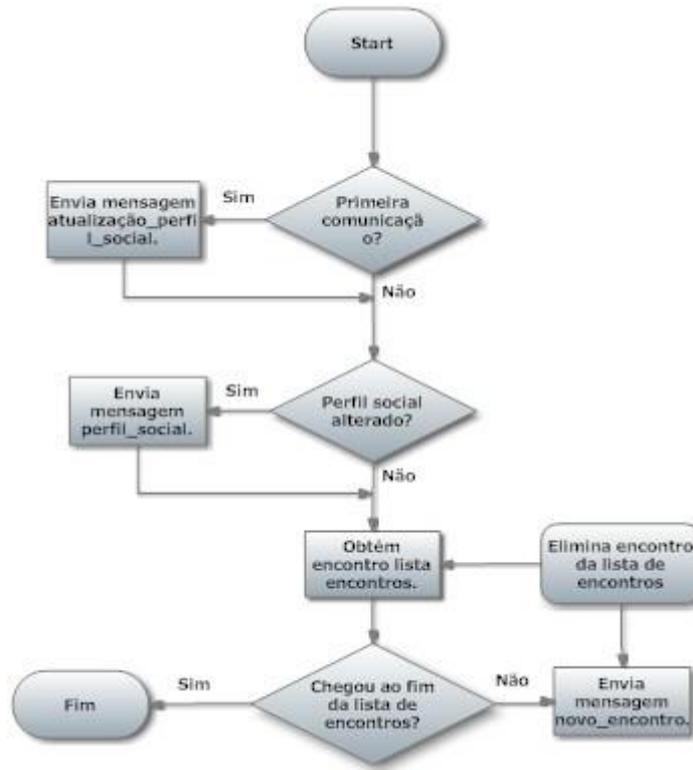


FIGURA IV.7 - LIGAÇÃO APLICAÇÃO-SERVIDOR

Cada vez que o utilizador acede à aplicação, esta tenta, através do uso de Services em Android em background, conectar-se ao servidor. Na Figura IV.7 está representada a forma como são enviadas as mensagens para o servidor.

## 4.2 Servidor

O servidor é um programa desenvolvido em Java que deve registar o servidor na Internet, para que este esteja acessível aos utilizadores através de um IP único pré-definido. O servidor funciona num Socket SSL com uso de certificados e chaves privadas.

O servidor recebe os dados dos clientes e constrói um grafo Neo4j, onde são guardados permanentemente os dados obtidos para posterior análise. Com os valores obtidos do histórico de contactos e dos perfis sociais, o servidor deve construir um grafo. Esse grafo, deve representar os nós e as ligações entre estes, relacionando os contactos físicos com o perfil social de cada utilizador. Uma ligação entre dois nós do grafo deve parecer-se com o da Figura IV.8. O servidor deverá, também, guardar para cada nó a seguinte informação: identificador do nó,

idade, número total de contactos com todos os outros nós, duração total dos contactos com todos os outros nós, lista de amigos e lista de interesses.

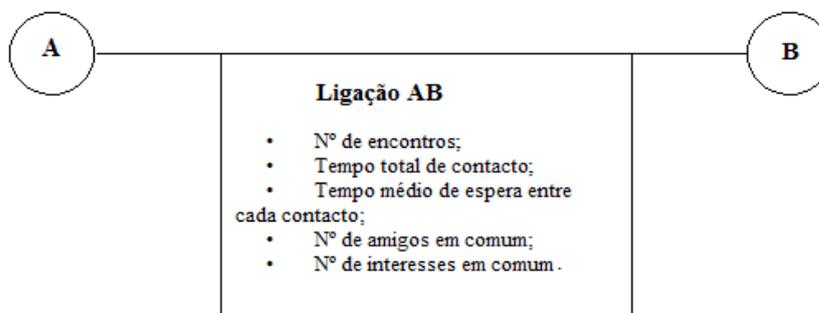


FIGURA IV.8 - LIGAÇÃO GRAFO SERVIDOR

Como já foi indicado em 4.1.5, o Servidor pode receber três tipos de mensagens: novo\_encontro, perfil\_social ou actualização\_perfil\_social.

Se receber uma mensagem perfil\_social, o servidor cria um novo nó na base de dados noSQL, com os dados recebidos na mensagem. A informação relativa aos contactos físicos deve ser inicializada a zero.

Se receber uma mensagem do tipo actualização\_perfil\_social, o servidor obtém o nó a que a mensagem se refere e atualiza a lista de interesses e/ou amigos com os dados presentes na mensagem.

Para fazer a gestão dos encontros entre nós, o servidor guarda localmente uma lista com os encontros entre todos os pares de nós do qual já recebeu informação acerca dos contactos físicos. Esta lista deve ser identificada pelo identificador do encontro, que deve ser único para cada par de nós. Se o servidor receber uma mensagem do tipo novo\_encontro, é obtido o identificador do utilizador que enviou a mensagem e do utilizador encontrado, e verifica-se se existe já algum encontro com o mesmo identificador na lista de encontros deste par de nós. Se o identificador já existir na lista, deve-se ao facto de o servidor já ter recebido a informação do mesmo encontro do outro nó. Por isso, neste caso, verifica-se qual dos dois encontros tem um tempo de duração maior e esse encontro é adicionado à lista e à base de dados. Se o encontro não estiver contido na lista de encontros do par de nós, este é adicionado à lista, mas não à base de dados. Apenas será adicionado à base de dados quando receber uma informação do outro nó a indicar a existência do mesmo encontro ou caso não receba um aviso do mesmo

encontro durante um dia. Depois da receção de cada mensagem, o servidor deve enviar uma mensagem de confirmação para os clientes.

Para cada ligação: o tempo total de contacto é calculado somando todos os tempos de contactos entre os dois nós; o número de contactos é obtido somando todos os contactos entre os nós; a lista de amigos em comum e a lista de interesses em comum é obtida através da sobreposição das listas de amigos e interesses de cada nó; o tempo médio de espera é obtido percorrendo a lista de contactos entre o par de nós e verificando qual o tempo médio entre contactos; por fim, o tempo decorrido entre contactos é atualizado a cada cinco minutos passados no servidor, sem que tenha recebido informação acerca dos contactos entre os nós.

## **Sumário**

Neste capítulo foi explicado como foram implementados a aplicação SocialConnector e o servidor. No próximo capítulo, será feita uma simulação da análise de dados com a utilização de um *dataset* já existente para verificar a validade dos coeficientes propostos.



## CAPÍTULO V TESTES E RESULTADOS

Nos capítulos anteriores, é explicado como são obtidos os dados do histórico de contactos físicos e do perfil social dos nós numa RTA. Neste capítulo, iremos analisar os dados com base em grafos e medidas de centralidade dos nós nesses grafos. Serão construídos quatro grafos para comparação: um grafo a exemplificar o histórico de contactos físicos entre os nós, um grafo a identificar a relação social entre os nós, um grafo com base no coeficiente físico apresentado na Equação 28, e outro grafo com base no coeficiente social apresentado na Equação 24. O propósito é verificar se os coeficientes apresentados são um bom indicador da relação física e social entre os nós da RTA. Para tal, será verificada a centralidade dos nós nos quatro grafos e a correlação entre o valor das centralidades calculadas entre o grafo de histórico de contactos físicos e o grafo do coeficiente físico; e entre o grafo social e o grafo do coeficiente social. A centralidade dos nós irá ser calculada em três tipos de métricas distintas: o seu grau, que mede o número de ligações que o nó possui; a centralidade *Betweenness*, que mede o número de vezes que o nó pertence aos caminhos mais curtos entre todos os nós da rede[30]; e a centralidade *Closeness*, que mede a distância média entre o nó e todos os outros nós da rede. Por fim, será calculada a métrica Amizade da Equação 31 para medir a relação entre o histórico de contactos físicos e a afinidade social entre todos os nós da rede RTA.

O objetivo seria recolher efetivamente os dados, instalando a aplicação SocialConnector num conjunto de dispositivos que seriam utilizados por um grupo de pessoas previamente escolhidas. Os dados obtidos seriam comparados com outros *datasets*. No tempo disponível para a realização deste trabalho não foi possível recolher estes dados devido à grande quantidade de tempo necessário para desenvolver uma experiência desta natureza, visto que seria fundamental recolher um número significativo de voluntários que aceitassem fazer parte da experiência. Esse grupo deveria ser cuidadosamente escolhido, de modo a obter resultados que nos permitissem tirar conclusões pertinentes acerca da relação entre o seu histórico de contactos físicos e a sua relação social.

Como tal não foi possível, utilizando o Crawdad [31], foi encontrado um *dataset* que permite avaliar o nosso sistema, o *dataset sigcomm2009* [32]. Optou-se por utilizar este *dataset*, pois possui o mesmo tipo de dados que se pretende para o nosso sistema, permitindo calcular as métricas apresentadas. Os dados do *dataset* irão ser analisados com os grafos referidos,

sendo que, futuramente, será necessário comparar estes dados com os dados obtidos através do sistema descrito nesta tese científica.

### **5.1 O Dataset Sigcomm09**

O *dataset sigcomm09* contém dados coletados de uma aplicação desenvolvida para telemóveis com o sistema operativo Windows Phone: a MobiClique [5]. A aplicação foi utilizada por 76 pessoas durante os cinco dias da conferência SIGCOMM, em 2009, em Barcelona. Para tal, durante os primeiros dois dias da conferência, foram fornecidos cerca de cem telemóveis, com a aplicação MobiClique instalada, a um conjunto de voluntários.

Os telemóveis entregues foram HTC s620 Windows Mobile, com um processador de 200MHz, RAM de 64Mb, ROM de 128Mb e espaço para colocar cartão de memória MicroSD. Os dispositivos entregues aos voluntários foram inicialmente carregados com o perfil social do voluntário, incluindo alguma informação básica, como o instituto a que pertenciam, a cidade e o país de origem. Foi também solicitado aos participantes uma autorização para obter a lista de amigos e de interesses do Facebook e incluir estas listas no perfil social. Todos os participantes foram adicionados a três grupos de interesses pré configurados. Antes do perfil social ser carregado no dispositivo e, conseqüentemente, ser guardado no *dataset*, cada participante podia alterar o seu perfil social de modo a não revelar informações que não pretendesse mostrar. Durante a experiência, foi possibilitado aos participantes criarem novas amizades com os outros participantes, bem como criarem novos interesses ou juntarem-se a interesses criados durante a experiência por outros utilizadores. Aos participantes, foi-lhes pedido que mantivessem o dispositivo fornecido ligado, e que o transportassem durante a conferência com a aplicação MobiClique ativa para troca de perfis sociais com os restantes participantes na rede. A rede utilizada foi uma rede oportunista baseada em Bluetooth, sendo que cada nó da rede correspondia a uma pessoa, e ao seu dispositivo. Cada dispositivo executa uma procura de dispositivos Bluetooth, com duração de 10.24 segundos, a cada 130.24 ou 109.76 segundos, com o propósito de descobrir dispositivos que se encontrem no seu raio de alcance. A cada contacto entre dois participantes, é gerada uma ligação RFCOMM num canal pré configurado para comunicação entre eles. Cada dispositivo grava em ficheiros de texto no seu cartão de memória o seu perfil social e os resultados da descoberta de novos dispositivos, bem como as

comunicações existentes entre eles. Com o intuito de manter a privacidade de cada participante, todos os identificadores obtidos através do Facebook, a informação pessoal de cada participante e o endereço MAC do Bluetooth foram substituídos por identificadores aleatórios. Todos os relógios dos dispositivos foram sincronizados com a mesma hora, e todos os valores temporais apresentados no dataset estavam baseados no relógio dos dispositivos, referindo-se ao tempo relativo em segundos, decorrido desde o início da experiência, 17/08/2009 08:00.

No final da experiência, todos os dados guardados pelos dispositivos foram agrupados em dez ficheiros do tipo CSV (Comma Separated Values) [33]. De seguida, apresentam-se os ficheiros utilizados para a construção das bases de dados NoSQL.

O ficheiro *Participants* reúne a informação social dos participantes, o instituto a que pertencem, a sua cidade e país de origem. Cada linha deste ficheiro contém os seguintes campos: identificador do participante, único em toda a experiência; uma chave; e um valor referente à instituição, cidade e país de origem. O valor dos campos é conseguido, através da atribuição de um identificador único e aleatório, permitindo, assim, manter a privacidade dos participantes da experiência. A chave pode ser instituto, cidade ou país.

O ficheiro *Activity*, onde estão guardados os tempos de início e fim de cada período de atividade de cada dispositivo, ou seja, os períodos de tempo em que o dispositivo está ligado e com a aplicação MobiClique ativa. Cada linha deste ficheiro contém os seguintes campos: identificador do participante, tempo relativo ao início da atividade e tempo relativo ao fim da atividade.

O ficheiro *Friends1* contém a lista de amizades entre os participantes. Cada linha deste ficheiro inclui os seguintes campos: identificador do participante e identificador do participante do qual é amigo. O ficheiro *Friends2* coleta a lista de amizades entre os participantes, estabelecidas durante a experiência. Cada linha deste ficheiro contém os seguintes campos: identificador do participante, identificador do participante do qual é amigo e o instante de tempo relativo ao início da amizade.

O ficheiro *Interests1* engloba a lista de interesses de cada participante. Cada linha deste ficheiro inclui os seguintes campos: identificador do participante, identificador do interesse. O ficheiro *Interests2* contém a lista de interesses a que os participantes se juntaram ou criaram durante a experiência. Cada linha deste ficheiro possui os seguintes campos: identificador do

participante, identificador do interesse e o instante de tempo relativo à junção do participante a este interesse.

O ficheiro *Proximity* contém os logs de cada dispositivo Bluetooth descoberto por cada participante. Cada linha deste ficheiro abrange os seguintes campos: identificador do participante, identificador do participante encontrado, instante da descoberta do dispositivo e os códigos da classe de Bluetooth do dispositivo.

Neste *dataset* existem três ficheiros mais que contêm informação acerca da comunicação efetuada entre os nós da rede: *Transmission*, *Reception*, e *Messages*. Como os dados presentes nestes ficheiros não serão utilizados na análise efetuada, estes não estão descritos acima.

Na Tabela V.1 podemos verificar os valores globais acerca dos encontros e dos perfis sociais obtidos nesta experiência, sendo que para o caso de estudo, tal como descrito em [6], consideramos um contacto entre um nó A e B como o tempo decorrido entre a primeira resposta de B à procura de dispositivos Bluetooth de A, até à inexistência de resposta de B entre dois períodos consecutivos de procura.

<b>Número de participantes</b>	76
<b>Duração da experiência</b>	5 Dias
<b>Total de interesses</b>	711
<b>Encontros entre os nós</b>	21926
<b>Média contactos entre nós</b>	$\approx 289,16$
<b>Média interesses por nó</b>	$\approx 12$
<b>Média amizades por nó</b>	$\approx 4$
<b>Média Duração contacto</b>	$\approx 222.49s$

TABELA V.1 - DATASET SIGCOMM09

Para simular a parte do servidor foi desenvolvido um programa em Java que lê os ficheiros *csv* do *dataset* escolhido e carrega os dados recolhidos em diferentes grafos Neo4J para analisar posteriormente. A análise dos grafos foi feita, utilizando a plataforma Gephi [34] [35] com o plugin Neo4J [36] instalado.

### 5.1.1 Preparação e Carregamento dos Dados

Foi, portanto, desenvolvido um programa em Java para a construção dos grafos que começa por carregar os dados relativos aos participantes da experiência, usando para tal, os ficheiros `participants.csv`, `activity.csv`, `friends1.csv`, `friends2.csv`, `interests1.csv` e `interests2.csv`. Com a informação recolhida, é formada uma lista de participantes com os seguintes campos: identificador do participante; identificador da instituição a que pertence; identificadores da sua cidade e país de origem; lista de amigos; lista de interesses e tempo de atividade. Através do ficheiro `proximity.csv`, obteve-se a lista de encontros entre os nós da rede. Neste ficheiro estão os encontros com dispositivos que não fazem parte da experiência, sendo que, na leitura do ficheiro, esses encontros são ignorados. A obtenção da lista de encontros de cada nó foi obtida da forma representada na Figura V.1, onde o valor 260 corresponde a duas procuras de dispositivos ( $2 \times (120 + 10.24)$ ). Depois de obtidos os contactos de cada nó, foi necessário normalizar a lista de contactos, visto que a informação que existia fazia parte da lista de encontros de cada nó, tendo havido, por isso, encontros repetidos. Devido ao desfasamento entre os valores temporais de cada dispositivo, foi necessário elaborar um algoritmo que, através da lista de contactos do nó A com o nó B e do nó B com um nó A, formasse uma só lista normalizada com os encontros efetivos entre os dois nós. O algoritmo elaborado consiste na eliminação de contactos que se sobrepõem, isto é, se na lista de contactos do nó A com o nó B existir um encontro que se inicie no instante de tempo  $t_1$  e acabe no instante de tempo  $t_2$ , e na lista de encontros do nó B com o nó A existir um encontro que se sobreponha, ou seja, que o seu início ou fim seja entre o instante  $t_1$  e  $t_2$ , verifica-se qual dos dois encontros tem a maior duração e adiciona-se o mesmo à lista de encontros normalizados entre o nó A e B. Na Figura V.2 está demonstrado um exemplo do funcionamento deste algoritmo.

Depois de carregados os dados para o computador, foram construídos dois grafos para análise: um grafo que demonstra o histórico dos contactos físicos entre os nós e um grafo que demonstra a sua relação social.

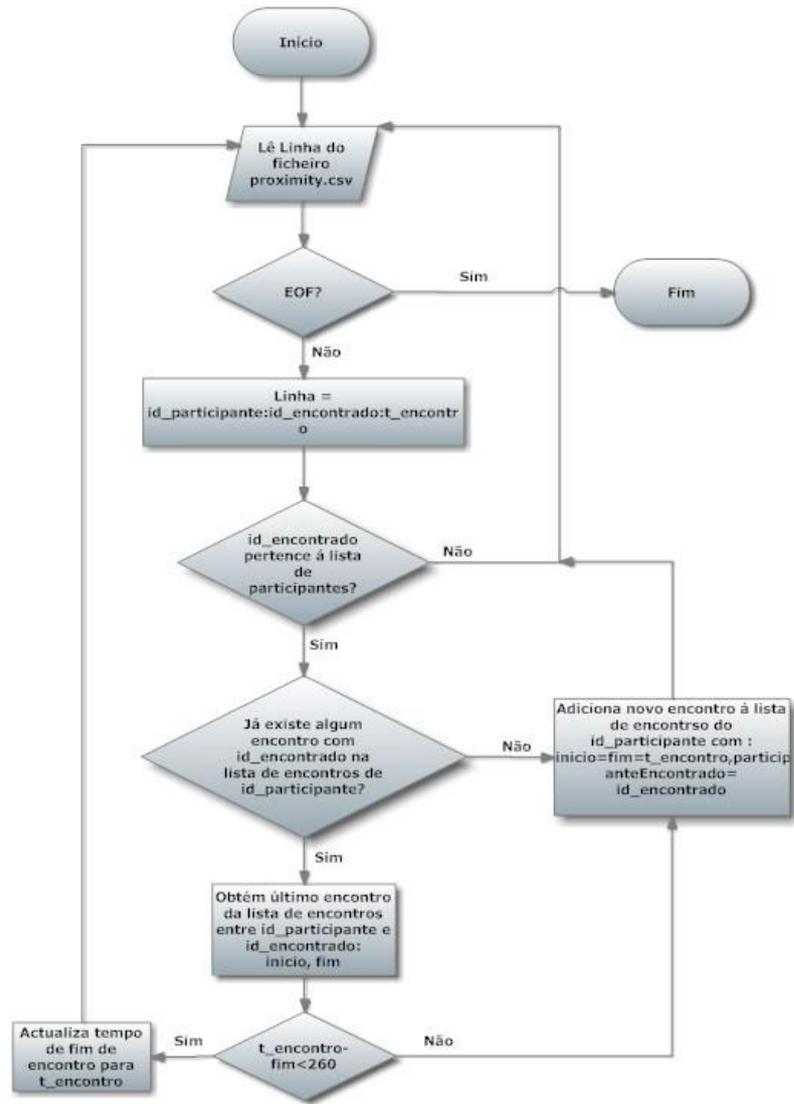


FIGURA V.1 - OBTENÇÃO CONTACTOS

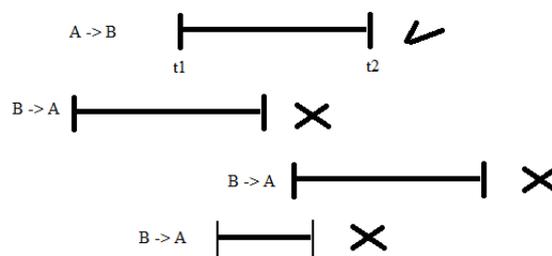
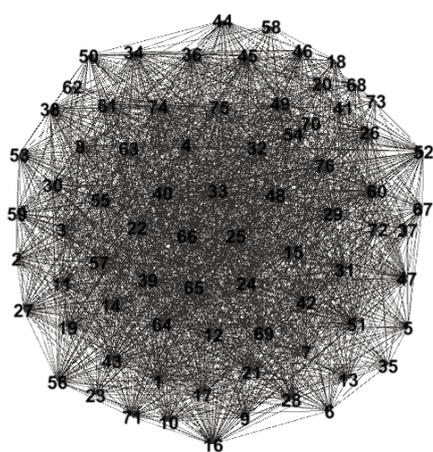


FIGURA V.2 - NORMALIZAÇÃO CONTACTOS

## 5.2 Grafo Histórico Contactos Físicos

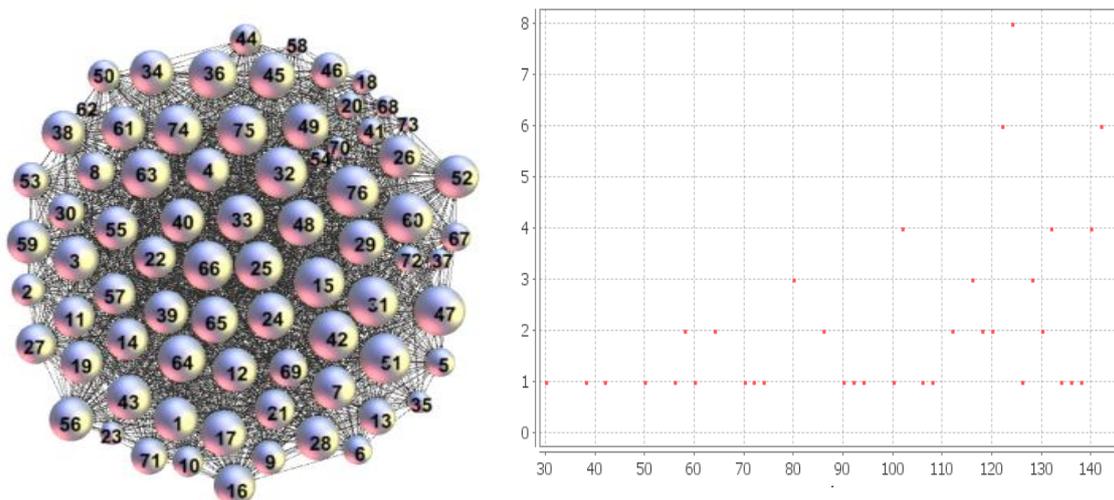
Para construir o grafo do histórico de contactos físicos entre os nós foi definido um tipo de ligação Neo4J: ENCONTRO. Os nós foram carregados na base de dados Neo4J com os seguintes atributos: identificador da instituição a que pertence, identificador da cidade de origem, identificador do país de origem e o tempo de atividade. Depois de carregados os nós, foi necessário estabelecer as ligações entre eles, sendo que esta apenas era estabelecida se houvesse pelo menos um contacto entre os nós na rede. As ligações numa base de dados Neo4J são unidireccionais. Pretende-se, assim, que uma ligação entre o nó A e B represente o histórico de contactos físicos entre os dois nós, e este deve ser igual nas duas direções, visto que se considera que se o nó A encontrou o nó B num determinado instante, o contrário também se verifica, ou seja, o nó B também encontrou o nó A nesse instante. Por isso, as ligações no grafo vão ser sempre bidireccionais, no momento em que se estabelece uma ligação do nó A ao nó B, estabelece-se uma ligação com as mesmas características mas na direção contrária. Uma ligação neste grafo tem as seguintes propriedades: tempo total de contactos, número total de contactos e tempo médio de espera entre contactos. Estes valores foram calculados pelo programa Java desenvolvido. Pode observar-se o grafo obtido na Figura V.3. A posição dos nós foi obtida usando o *layout Force Atlas* [34].



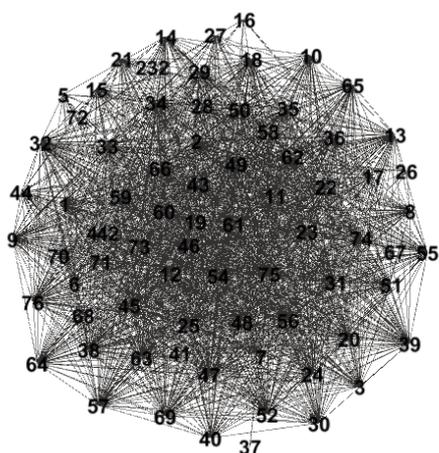
Nº de nós	76
Nº de Ligações	4164

FIGURA V.3 - GRAFO FÍSICO

Através do Gephi, foi calculado o grau de cada nó. Esta medida indica o número de ligações de cada nó, medindo, assim, a importância do nó na rede. Usaremos, então, esta medida como a medida principal para medir a centralidade dos nós da rede. No gráfico de frequências da Figura V.4, observa-se a distribuição do grau pelos nós da rede. No grafo da Figura V.4 observamos o mesmo grafo da Figura V.3 mas com a variação do tamanho de cada nó a ser diretamente proporcional ao seu número de ligações. A partir deste grafo, podemos concluir que, para esta experiência, os dez nós mais centrais em termos de contactos físicos são os seguintes: 32,76,15,31,47,51,60,75,42 e 63. Já os nós menos centrais no grafo são: 62,73,58,54,70,37,68,35,18 e 23. O valor máximo obtido foi de 146 para os nós 32 e 76 e o mínimo de 30 para o nó 62.



participantes da experiência foram automaticamente adicionados a três grupos, e como essa informação é irrelevante e igual para todos os nós, foi decidido não considerar esses três grupos- e a ligação AMIGOSCOMUM existe, se os dois nós tiverem pelo menos um amigo em comum. Todas estas ligações são bidirecionais, ou seja, se o nó A é amigo de B, considera-se B amigo de A, o mesmo acontecendo para os amigos e interesses em comum. A ligação AMIGODE não tem nenhuma propriedade, as ligações AMIGOSCOMUM e INTERESSESCOMUM têm como propriedade o número de amigos e interesses em comum respectivamente. A posição dos nós foi obtida da mesma forma que o grafo físico, ou seja, recorrendo ao *layout* Force Atlas. Na Figura V.5 podemos observar o grafo obtido.



Nº de nós	76
Nº de Ligações	3160

FIGURA V.5 - GRAFO SOCIAL

Utilizando o Gephi, é possível obter a percentagem das relações de cada tipo, das 3160 ligações existentes: 44,43% correspondentes a 1404 são ligações do tipo INTERESSESCOMUM, 41.84% correspondentes a 1322 ligações são do tipo AMIGOSCOMUM e somente 13.73% correspondentes a 434 ligações são do tipo AMIGODE. Através destes valores, podemos concluir que os nós do grafo social não têm muitas relações diretas de amizade entre eles, mas por outro lado, existe uma relação entre os seus amigos e interesses.

Da mesma forma que foi feito o cálculo do grau de cada nó para o grafo físico, o mesmo aconteceu para o grafo social, com o intuito de verificar a centralidade dos nós no grafo e, posteriormente, usar os valores obtidos para comparação. O gráfico com a distribuição dos valores do número de ligações em cada nó pode ser observado na Figura V.6. No grafo da

Figura V.6 podemos observar o grafo da Figura V.5 com o tamanho dos nós a ser proporcional ao seu grau. Os nós mais centrais do grafo social são: 34, 47, 32, 45, 22, 61, 76, 39, 52 e 13. Já os nós com um menor valor de centralidade são: 53, 37, 72, 42, 41, 16, 19, 26, 28 e 67. O valor de grau mais elevado, 138, pertence ao nó 34, já o valor mais baixo, ou seja 0, pertence ao nó 53.

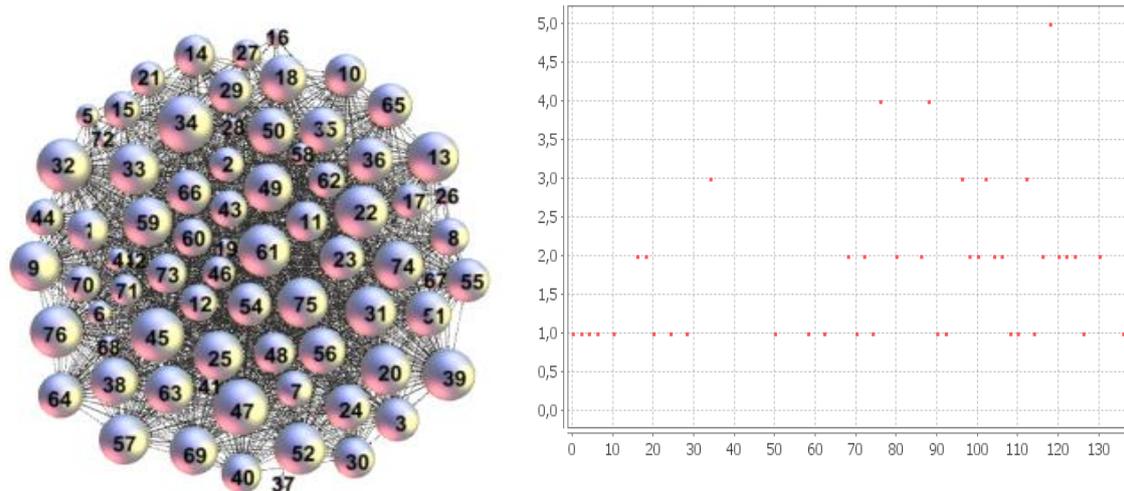


FIGURA V.6 - DISTRIBUIÇÃO GRAU DOS NÓS GRAFO SOCIAL

## 5.4 Grafo Coeficiente Físico

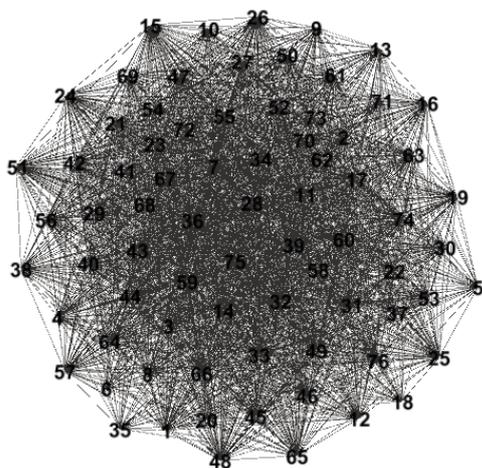
Com o propósito de verificar a validade das métricas apresentadas nesta tese para relacionar os dados de histórico de contactos físicos entre dois nós com a sua relação social, foram construídos dois grafos usando os coeficientes das Equações 24 e 28. Estas métricas pretendem atribuir um valor entre 0 e 1 à relação social e física entre os dois nós da rede. O objetivo final da construção dos grafos utilizando estas métricas é verificar se existe uma relação direta com os grafos social e físicos, representados na Figura V.3 e Figura V.5.

Para a construção do grafo que relaciona os nós através do histórico de contactos físicos, foi definido um tipo de ligação: COEFFISICO. Este tipo de ligação tem como propriedade o coeficiente físico, valor entre 0 e 1 calculado na Equação 28. Foram definidos o valor de 0.5 para  $\delta$  e  $\alpha$ , atribuindo, assim, a mesma importância ao número e duração dos contactos. O valor escolhido para  $\rho$  foi de 0.98, e o tempo médio de espera entre os nós foi convertido de

segundos para horas após se ter verificado que utilizando segundos os valores dos coeficientes seriam mínimos.

A ligação COEFFISICO apenas é estabelecida entre dois nós da rede, se o valor calculado do coeficiente físico for superior a um valor definido de *threshold* de 0.01. Este valor de *threshold* foi escolhido após a realização de vários testes, em que se verificou que este valor era o mais apropriado para verificar a existência de uma relação física entre dois nós, pois devido à constante de envelhecimento no cálculo do coeficiente físico, os valores obtidos são muito baixos, tendo um valor máximo de 0.48, um valor mínimo de 0 e um valor médio de 0.01008. Este valor indica, portanto, que somente consideramos que existe uma relação física relevante entre A e B, se o coeficiente físico for superior a 0.01.

Visto que o valor do coeficiente físico depende do número total de contactos e da sua duração em cada nó individualmente, o valor do coeficiente será diferente consoante estejamos a calcular a relação do nó A com o nó B ou do nó B com o nó A. Por isso, as ligações neste grafo não são bidirecionais, ou seja, pode existir uma ligação entre o nó A e o nó B e o contrário não se verificar. O grafo obtido pode ser verificado na Figura V.7.



Nº de nós	76
Nº de Ligações	3801

FIGURA V.7 - GRAFO COEFICIENTE FISICO

Tal como para os grafos do histórico de contactos físicos e social, foi calculado o grau de cada nó do grafo do coeficiente físico. O gráfico de ocorrências do número de ligações por nó pelos nós do grafo está representado na Figura V.8. O grafo da Figura V.8 refere-se ao grafo da

Figura V.7 mas com o tamanho dos nós a ser proporcional ao número de ligações que o nó possui. Analisando a Figura V.8, podemos observar que os nós mais centrais da rede são os nós: 15, 32, 31, 76, 36, 42, 60, 33, 51 e 48. Já os nós com um menor valor de centralidade são: 62, 73, 58, 54, 37, 35, 70, 20, 68 e 23. O valor máximo obtido foi de 137 para os nós 15 e 32, e o mínimo de 25 para o nó 62.

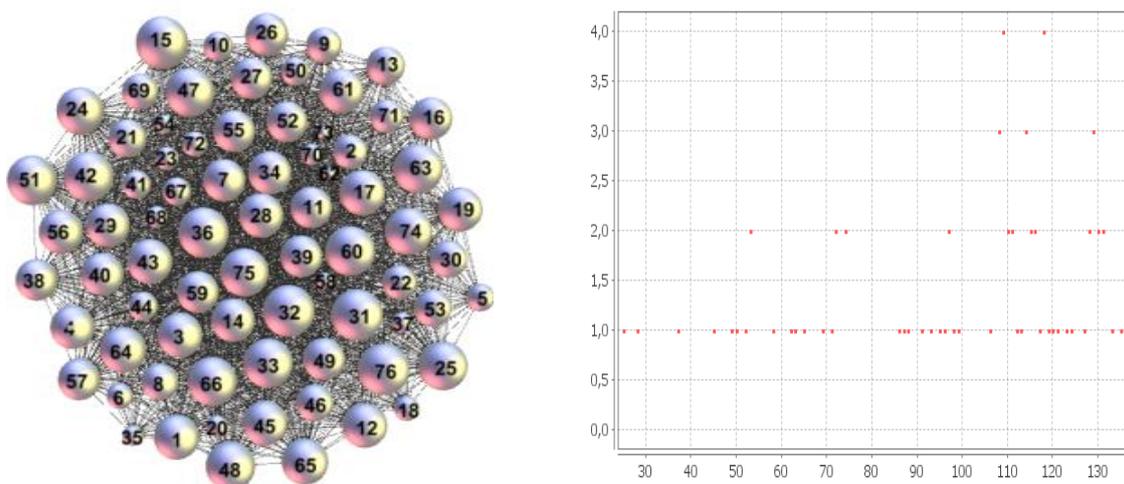


FIGURA V.8 - DISTRIBUIÇÃO GRAU DOS NÓS GRAFO COEFICIENTE FÍSICO

### 5.4.1 Comparação com Grafo Físico

Na Tabela V.2 podemos observar os valores do grau, da centralidade Betweenness e da centralidade Closeness, de cada nó para os dez nós mais centrais e menos centrais em termos de grau, do grafo de coeficiente físico e do grafo do histórico de contactos físicos.

Foi calculada a correlação entre os valores das centralidades nos dois grafos. Na Tabela V.3 estão demonstrados os valores obtidos.

<b>Nó</b>	<b>Grau dos nós Grafo Coeficiente Físico</b>	<b>Grau dos nós Grafo Histórico Contactos Físicos</b>	<b>Centralidade Betweenness Grafo Coeficiente Físico</b>	<b>Centralidade Betweenness Grafo Histórico Contactos Físicos</b>	<b>Centralidad e Closenness Grafo Coeficiente Físico</b>	<b>Centralidad e Closenness Grafo Histórico Contactos Físicos</b>
32	137	146	55.885	53.247	1.133	1.027
76	135	146	57.304	58.525	1.16	1.027
15	137	142	56.774	46.84	1.12	1.053
31	136	142	51.653	41.243	1.133	1.053
47	128	142	37.165	42.179	1.24	1.053
51	130	142	57.689	44.78	1.173	1.053
60	131	142	65.115	50.857	1.147	1.053
75	127	142	48.105	49.948	1.2	1.053
42	131	140	53.406	42.406	1.173	1.067
63	129	140	51.091	48.939	1.173	1.067
36	133	138	67.767	48.945	1.147	1.08
33	130	132	40.276	28.576	1.147	1.12
48	129	132	45.925	36.095	1.16	1.12
62	25	30	0.694	0.472	1.8	1.8
73	28	38	0.504	0.474	1.747	1.747
58	37	42	0.367	0.105	1.72	1.72
54	45	50	7.038	4.811	1.667	1.667
70	52	56	1.439	0.327	1.439	1.627
37	49	58	3.626	2.856	1.613	1.613
68	53	58	4.667	2.94	1.613	1.613
35	50	60	2.867	1.904	1.6	1.6
18	62	64	6.209	3.367	1.573	1.573
23	58	64	2	1	1.573	1.573
20	53	70	4.334	3.886	1.56	1.533

TABELA V.2 - COMPARAÇÃO CENTRALIDADES DOS NÓS GRAFO HISTÓRICO CONTACTOS FÍSICO E GRAFO COEFICIENTE

FÍSICO

<b>Centralidade</b>	<b>Correlação</b>
Grau	0,995883868
Betweenness	0,977630754
Closeness	0,980075894

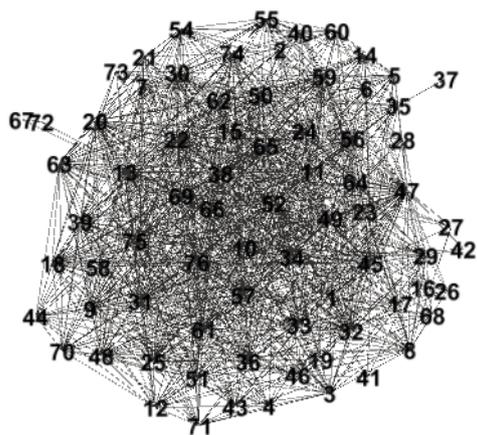
TABELA V.3 - CORRELAÇÃO VALOR CENTRALIDADE GRAFO COEFICIENTE FÍSICO E GRAFO HISTÓRICO DE CONTACTOS FÍSICOS

Da análise da Tabela V.3, podemos retirar que o coeficiente físico apresentado pode ser usado para representar o histórico de contactos físicos entre os nós numa RTA. A pequena discrepância entre os valores das centralidades deve-se ao facto de que o grafo do coeficiente físico inclui uma relação entre o número e duração dos contactos, ao contrário do grafo de histórico de contactos físicos em que só é tido em conta o encontro efetivo entre os nós, e por isso, no grafo coeficiente físico, as ligações irrelevantes são retiradas.

## 5.5 Grafo Coeficiente Social

Da mesma forma que foi construído um grafo com os valores de coeficientes físicos obtidos para cada ligação, foi realizado, igualmente, um grafo com as ligações a corresponder ao valor do coeficiente social calculado na Equação 24, com a importância dada aos três tipos de coeficientes igual, e por isso  $\vartheta = \sigma = \beta = 0.33$ . Tal como no grafo do coeficiente físico, este grafo tem somente um tipo de ligação: COEFICIENTESOCIAL.

A ligação entre dois nós apenas é estabelecida se o valor do coeficiente social ultrapassar o valor de *threshold* definido de 0.1. O valor de *threshold* definido para este grafo é superior ao definido para o grafo do coeficiente físico, pois devido à constante de envelhecimento, os valores do coeficiente físico são muito inferiores ao coeficiente social, que depende unicamente da proporção dos interesses e amigos em comum entre dois nós. O valor máximo do coeficiente social obtido para uma ligação entre dois nós foi de 0.852, o valor mínimo de 0 e o valor médio de 0.103. O grafo obtido fazendo depender as ligações entre os nós do seu coeficiente social é apresentado na Figura V.9.



Nº de nós	76
Nº de Ligações	1715

FIGURA V.9 - GRAFO COEFICIENTE SOCIAL

Tal como para os grafos anteriores, foi calculado o grau de cada nó. O gráfico de ocorrências do número de ligações de cada nó para este grafo está representado na Figura V.10. De forma a fazer-se a comparação com o grafo social da Figura V.9, apresenta-se na Figura V.10 o mesmo grafo, mas com o tamanho dos nós a ser proporcional à sua centralidade na rede, que se baseia no número total de ligações. Os nós mais centrais neste grafo são: 47, 22, 45, 75, 34, 20, 31, 76, 52, 32, já os nós menos centrais são: 53, 37, 67, 72, 42, 16, 26, 41, 27, 73.

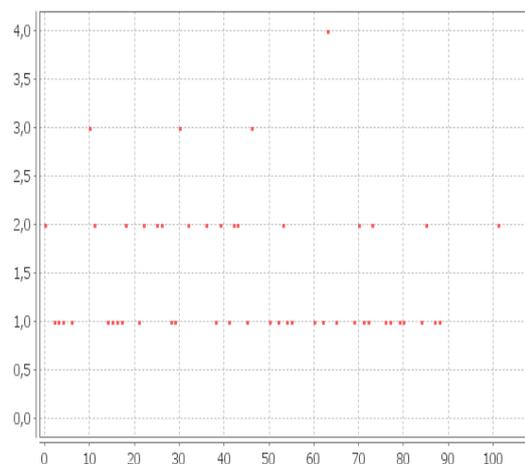
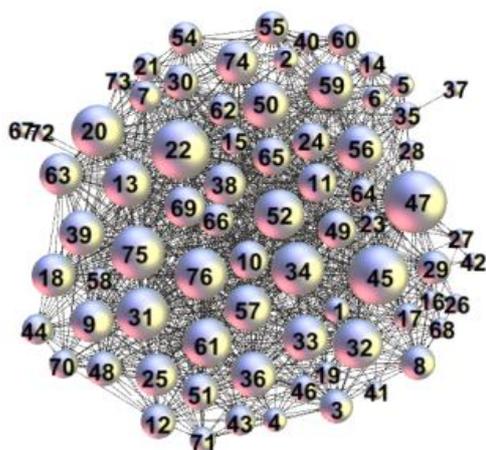


FIGURA V.10 - DISTRIBUIÇÃO GRAU DOS NÓS GRAFO COEFICIENTE SOCIAL

O valor mais elevado do grau de um nó obtido neste grafo é de 108 e pertence ao nó 47, o valor mais baixo é 0 e pertence ao nó 53. Tal como no grafo do coeficiente físico, as ligações são unidireccionais visto que o coeficiente social do nó A para o nó B pode ter um valor diferente do coeficiente social do nó B para o nó A.

### 5.5.1 Comparação com Grafo Social

Na Tabela V.5 podemos contrastar os valores do grau, da centralidade Betweenness e da centralidade Closeness para os treze nós mais e para os doze nós menos centrais no grafo social e no grafo do coeficiente social. Foi calculada a correlação entre os valores das centralidades nos dois grafos, na Tabela V.4, estão demonstrados os valores obtidos.

<b>Centralidade</b>	<b>Correlação</b>
Grau	0,948492344
Betweenness	0,698184553
Closeness	0,651151184

TABELA V.4 - CORRELAÇÃO VALOR CENTRALIDADE GRAFO SOCIAL E GRAFO COEFICIENTE SOCIAL

Pela análise da Tabela V.4, podemos concluir que o coeficiente social apresentado pode ser usado para representar relação social entre os nós numa RTA. A discrepância entre os valores das centralidades é maior que nos grafos físicos, pois, no grafo social, cada par de nós pode ter entre uma a três ligações: INTERESSESCOMUM, MIGOSCOMUM e AMIGODE; e, no grafo do coeficiente social, cada par de nós só pode ter no máximo uma: COEFICIENTESOCIAL. Para além disso, a discrepância também se deve ao facto de que o grafo do coeficiente social exclui ligações sociais fracas, ou seja, aquelas em que os nós têm poucos amigos e interesses em comum.

<b>Nó</b>	<b>Grau dos Nós Grafo Coeficiente Social</b>	<b>Grau dos nós Grafo Social</b>	<b>Centralidade Betweenness Grafo Coeficiente Social</b>	<b>Centralidade Betweenness Grafo Social</b>	<b>Centralidade Clossenness Grafo Coeficiente Social</b>	<b>Centralidade Clossenness Grafo Social</b>
47	108	136	456.392	185.715	1.473	1.081
22	101	126	278.075	56.165	1.446	1.149
45	101	130	429.672	58.237	1.5	1.122
75	88	118	151.701	64.498	1.486	1.203
34	87	138	184.224	222.642	1.689	1.068
20	85	118	101.933	32.461	1.486	1.203
31	85	118	115.801	36.988	1.459	1.203
76	84	124	107.632	75.964	1.635	1.162
52	80	122	148.39	70.857	1.581	1.176
32	79	130	110.68	98.267	1.568	1.122
61	77	124	96.708	58.237	1.703	1.162
39	69	122	56.283	37.506	1.716	1.176
13	76	120	51.385	34.312	1.662	1.189
53	0	0	0	0	0	0
37	2	2	0	0	2.946	2.284
67	3	20	0	12.934	2.527	1.905
72	4	4	72	0	2.527	2.203
42	6	6	0	0	2.284	1.986
16	10	16	0	0	2	1.932
26	10	18	0	0	2.041	1.905
41	10	10	0.367	0	2.189	1.959
27	11	50	1.138	0.589	1.138	1.689
73	11	86	0	1.61	0	1.419
19	16	16	2.15	0	2.095	1.919
28	15	18	0.609	0	0.609	1.905

TABELA V.5 - COMPARAÇÃO CENTRALIDADE DOS NÓS GRAFO SOCIAL E GRAFO COEFICIENTE SOCIAL

## 5.6 Cálculo Amizade entre os Nós

Depois de analisados e comparados os grafos sociais e físicos, foi calculada a métrica *Amizade* da Equação 31, para relacionar o campo físico e o campo social entre os nós da rede. Esta métrica calcula a razão entre o coeficiente físico e o coeficiente social e, consoante o valor obtido, pode concluir-se acerca da relação entre o histórico de contactos físicos e da afinidade social entre os nós. Se o valor da métrica *Amizade* for muito superior a 1, existe mais relação física que social entre os nós; se for próxima de 0 existe mais afinidade social que relação física; se for próxima de 1, tem uma afinidade social e uma relação física semelhante. Se o coeficiente físico for 0, o valor de *Amizade* é 0; se o coeficiente social for 0 a *Amizade* toma um valor incalculável (divisão por 0), e por isso, a estas ligações é atribuído o valor 100; por fim, se os dois coeficientes forem 0 a *Amizade* toma o valor -1. Na Figura V.11 podemos observar a dispersão dos valores obtidos da amizade para as 5700 relações entre todos os nós da rede.

Da análise da Figura V.11, podemos retirar as seguintes conclusões:

- Existe um maior número de ligações com *Amizade* perto do valor 0 do que perto do valor 100, o que nos indica que as relações entre nós têm uma afinidade social maior que a relação física;
- A maioria das ligações não tem nenhuma relação física (valores 0), nenhuma afinidade social (100) ou nenhuma relação (-1).
- Existe um número considerável de ocorrências entre os valores 0.25 e 0.5.
- Existem poucas relações com o valor próximo de 1.

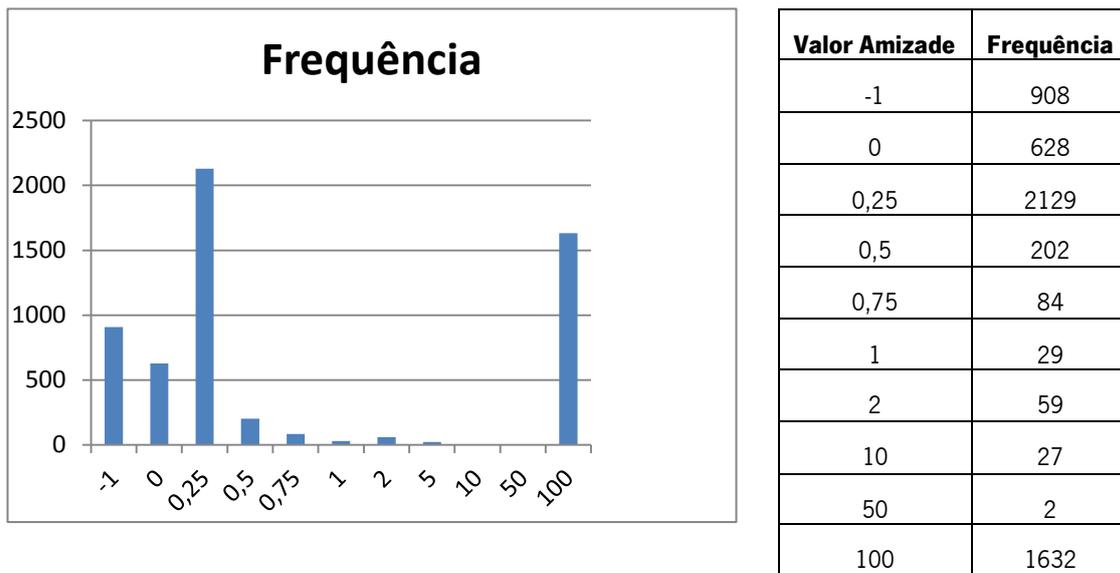


FIGURA V.11 - DISTRIBUIÇÃO AMIZADE

Os dados obtidos para o valor da *Amizade* indicam que a maioria das ligações não têm uma relação física próxima da sua afinidade pessoal, mas esta situação deve-se ao facto do *dataset* estudado ser muito específico, uma vez que foi realizado durante uma conferência onde normalmente existem contactos entre pessoas que não se relacionam socialmente. A maioria dos valores encontra-se entre o valor 0.25 e o valor 0.5, devido ao facto dos coeficientes físicos serem inferiores aos coeficientes sociais calculados. Esta situação é explicada pelo facto dos utilizadores poderem esconder parte do seu perfil social, ou seja, alguns dos seus interesses e amigos, o que leva a que a informação social recolhida não esteja completa, fazendo com que os coeficientes sociais calculados tenham um valor acima do que seria esperado.



## CAPÍTULO VI CONCLUSÕES E TRABALHO FUTURO

Nesta dissertação foi apresentado um sistema para recolher dados sociais e de contactos físicos entre pessoas numa RTA. Foram definidos dois coeficientes para relacionar os dois campos indicados, um coeficiente físico para medir a força da relação física entre dois nós e um coeficiente social para medir a afinidade social entre duas pessoas. O objetivo seria testar estes coeficientes como métricas a ser usadas na decisão de encaminhamento num protocolo RTA com encaminhamento social. Para obter os dados para o teste dos coeficientes, foi implementada uma aplicação - a SocialConnector - responsável por obter os dados do histórico de contactos físicos e o perfil social de cada nó e enviar essa informação para o servidor desenvolvido para o efeito. O servidor é responsável por receber os dados dos nós da RTA e guardar esses dados numa base de dados Neo4j. Depois de obtidos os dados no servidor, o objetivo seria realizar uma análise dos mesmos com os coeficientes apresentados. Com esse intuito, foi apresentada, igualmente, uma métrica, a métrica *Amizade*, para verificar se dois nós com uma relação física forte têm uma relação social igualmente forte, ou seja, se duas pessoas que se encontram frequentemente tendem a ter os mesmos interesses e os mesmos grupos de amigos. Na altura da escrita deste documento não foi possível realizar a experiência com o sistema construído. Futuramente, esse trabalho será desenvolvido. Como o objetivo seria comparar os dados obtidos com outros *datasets* para testar as métricas apresentadas, foi utilizado um *dataset* já existente, o *dataset sigcomm09*. Este *dataset* possui dados semelhantes aos pretendidos para o sistema, e por essa razão foi escolhido para testar as métricas. Com os dados sociais e físicos do *dataset*, foram construídos quatro grafos: um grafo com as ligações físicas entre os nós, um grafo com as ligações sociais entre os nós, um grafo com base no coeficiente físico apresentado e um grafo com base no coeficiente social apresentado. Comparando o grafo do histórico de contactos físicos com o grafo baseado no coeficiente físico e comparando o grafo social com o grafo baseado no coeficiente social, foi provada a validade dos coeficientes calculados através da comparação da centralidade dos nós em cada grafo. Por fim, foi calculada a métrica *Amizade* para todas as ligações entre os nós do *dataset*, e verificou-se que a afinidade entre os dados sociais e físicos não estava relacionada, ao contrário do que se esperava. Conclui-se, portanto, que estes resultados se devem ao contexto em que os dados

foram recolhidos, ou seja, numa conferência onde pessoas de diferentes locais e de diferentes grupos sociais se encontram frequentemente.

O trabalho a ser realizado futuramente consiste na realização de testes à aplicação e servidor desenvolvidos e à realização de uma experiência que permita recolher dados para a construção de um *dataset* completo, com várias pessoas de diferentes grupos sociais e com diferentes relacionamentos entre elas.

Outros dos aspetos a ser incrementado consiste no acrescento de uma constante de envelhecimento no coeficiente social de modo a poder ser comparável com o coeficiente físico, para isso, será necessário obter um histórico de interações sociais entre os utilizadores através da troca de mensagens ou *posts*.

Por fim, outro aspeto a ser testado no futuro, seria a utilização de um simulador de RTAs, como o The One[37], para a validação da proposta de decisão de encaminhamento apresentada nesta dissertação.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] K. Fall, "A delay-tolerant network architecture for challenged internets," *Proc. 2003 Conf. Appl. Technol. Archit. Protoc. Comput. Commun. - SIGCOMM '03*, p. 27, 2003.
- [2] L. Fratta, M. Gerla, and L. Kleinrock, "The flow deviation method: An approach to store-and-forward communication network design," *Networks, Vol. 3, No 2*, pp. 97-133, 1980.
- [3] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," *Proceeding 2005 ACM SIGCOMM Work. Delay-tolerant Netw. - WDTN '05*, pp. 244-251, 2005.
- [4] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of Human Mobility on Opportunistic Forwarding Algorithms," *IEEE Trans. Mob. Comput.*, vol. 6, no. 6, pp. 606-620, Jun. 2007.
- [5] A. Pietiläinen, E. Oliver, and J. LeBrun, "MobiClique: middleware for mobile social networking," *2nd ACM Work. Online Soc. networks*, pp. 49-54, 2009.
- [6] A. Pietiläinen and C. Diot, "Dissemination in opportunistic social networks: the role of temporal communities," *MobiHoc '12 Proc. Thirteen. ACM Int. Symp. Mob. Ad Hoc Netw. Comput.*, 2012.
- [7] T. Nicolai, E. Yoneki, N. Behrens, and H. Kenn, "Exploring social context with the wireless rope," *Move to Meaningful Internet Syst. 2006 OTM 2006 Work.*, pp. 874-883, 2006.
- [8] J. Bondy and U. Murty, "Graph theory (Graduate texts in mathematics, Vol. 244)," no. 2000, 2008.
- [9] A. Mtibaa, A. Chaintreau, J. LeBrun, E. Oliver, A.-K. Pietiläinen, and C. Diot, "Are you moved by your social network application?," *Proc. first Work. Online Soc. networks - WOSP '08*, p. 67, 2008.
- [10] A. Vahdat and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks," *Tech. Rep. CS-200006, Duke Univ.*, 2000.
- [11] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, p. 19, Jul. 2003.
- [12] E. M. Daly and M. Haahr, "Social Network Analysis for Information Flow in Disconnected Delay-Tolerant MANETs," *IEEE Trans. Mob. Comput.*, vol. 8, no. 5, pp. 606-621, May 2009.
- [13] J. M. Pujol, A. Lopez Toledo, and P. Rodriguez, "Fair Routing in Delay Tolerant Networks," *IEEE INFOCOM 2009 - 28th Conf. Comput. Commun.*, pp. 837-845, Apr. 2009.

- [14] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE Rap : Social-Based Forwarding in Delay-Tolerant Networks," *Mob. Comput. IEEE Trans.*, pp. 1576–1589, 2011.
- [15] A. Mtibaa and M. May, "Peoplerank: Social opportunistic forwarding," *INFOCOM 2010 Proc. IEEE*, 2010.
- [16] K. Scott, S. Burleigh, and The MITRE Corporation, "Bundle Protocol Specification," 2007.
- [17] L. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.
- [18] M. Everett and S. P. Borgatti, "Ego network betweenness," *Soc. Networks*, vol. 27, no. 1, pp. 31–38, Jan. 2005.
- [19] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: bringing order to the web.," 1999.
- [20] R. Balani, "Energy Consumption Analysis for Bluetooth , WiFi and Cellular Networks," *Online*. <http://nesl.ee.ucla.edu/fw/documents/reports/2007/PowerAnalysis.pdf> (2007).
- [21] B. Sig, "Specification of the Bluetooth system, Master Table of Contents and Compliance Requirements," 2004.
- [22] J. Haartsen, "Bluetooth-The universal radio interface for ad hoc, wireless connectivity," *Ericsson Rev.*, pp. 110–117, 1998.
- [23] "Utilizadores Redes Sociais." [Online]. Available: <http://www.ebizmba.com/articles/social-networking-websites>. [Accessed: 01-Nov-2013].
- [24] M. Khan, "SQL vs NoSQL," 2012.
- [25] K. Strauch, Christof Ultra-Large Scale Sites and Walter, "NoSQL Databases."
- [26] C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y. Chen, and D. Wilkins, "A comparison of a graph database and a relational database," in *Proceedings of the 48th Annual Southeast Regional Conference on ACM SE 10*, 2010, p. 1.
- [27] J. Webber, "A programmatic introduction to Neo4j," in *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity - SPLASH '12*, 2012, p. 217.
- [28] "APIs Bluetooth for Android." [Online]. Available: <http://developer.android.com/guide/topics/connectivity/bluetooth.html>. [Accessed: 30-Oct-2013].
- [29] D. Eastlake and P. Jones, "US secure hash algorithm 1 (SHA1)," 2001.

- [30] U. Brandes, "A faster algorithm for betweenness centrality\*," *J. Math. Sociol.*, vol. 25, no. 1994, pp. 163–177, 2001.
- [31] "Crawdad." [Online]. Available: <http://crawdad.cs.dartmouth.edu/index.php>. [Accessed: 31-Oct-2013].
- [32] "Dataset Sigcomm09." [Online]. Available: <http://crawdad.cs.dartmouth.edu/meta.php?name=thlab/sigcomm2009>. [Accessed: 31-Oct-2013].
- [33] Y. Shafranovich, "Common format and mime type for comma-separated values (csv) files," 2005.
- [34] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: an open source software for exploring and manipulating networks.," *ICWSM*, pp. 361–362, 2009.
- [35] "Gephi." [Online]. Available: <https://gephi.org/>. [Accessed: 31-Oct-2013].
- [36] "Plugin Neo4J Gephi." [Online]. Available: <http://gephi.org/tag/neo4j/>. [Accessed: 31-Oct-2013].
- [37] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," *Proc. Second Int. ICST Conf. Simul. Tools Tech.*, 2009.